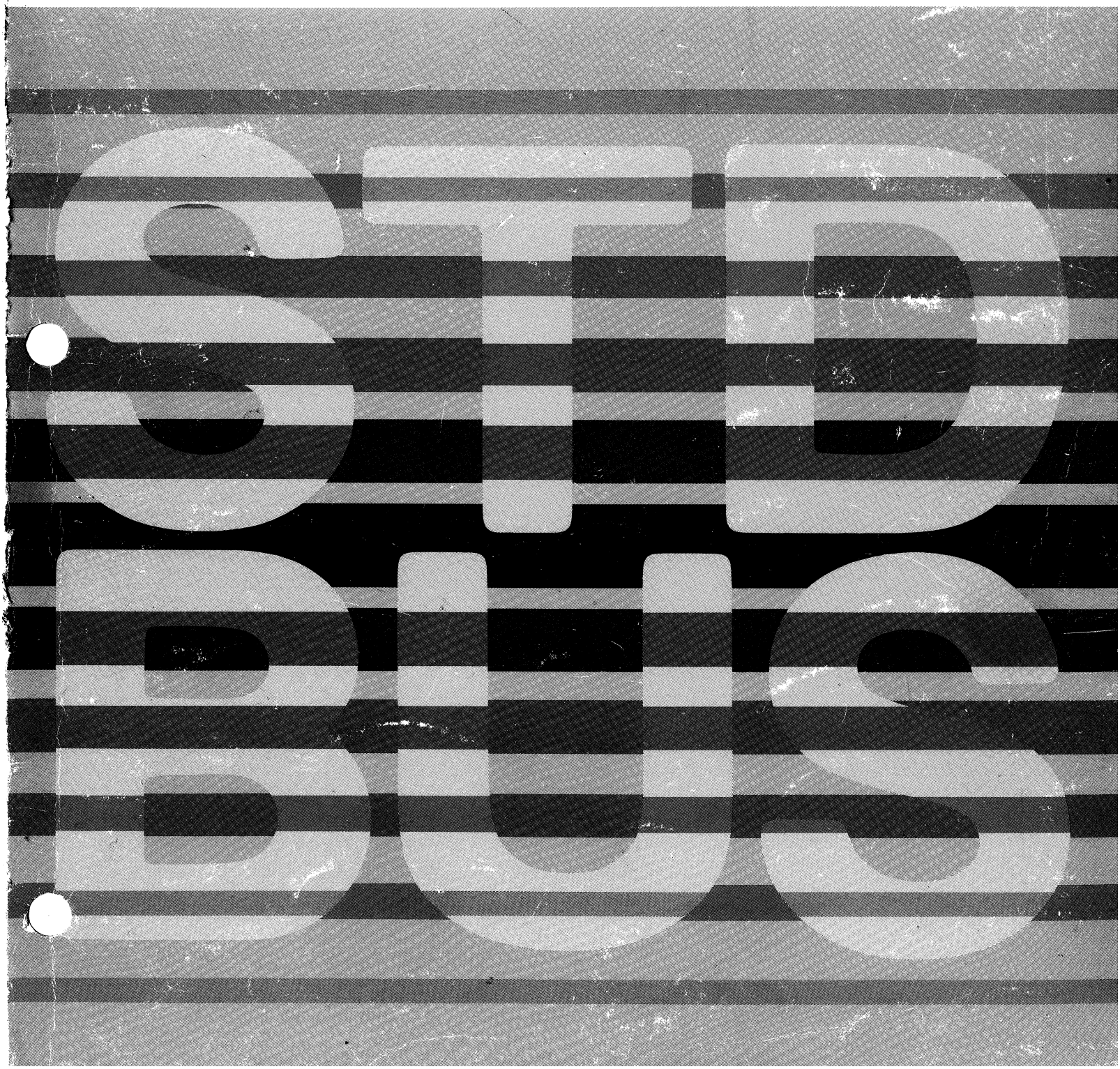
PRO-LOG®
CORPORATION

STD 7000

7842
Z80 RS-422 CPU Card

USER'S MANUAL



NOTICE

The information in this document is provided for reference only. Pro-Log does not assume any liability arising out of the application or use of the information or products described herein.

This document may contain or reference information and products protected by copyrights or patents and does not convey any license under the patent rights of Pro-Log, nor the rights of others.

Printed in U.S.A. Copyright © 1984 by Pro-Log Corporation, Monterey, CA 93940. All rights reserved. However, any part of this document **may be reproduced** with Pro-Log Corporation cited as the source.

STD 7000

7842 Z80A/RS-422

Multifunction CPU Card

USER'S MANUAL

FOREWORD

This manual explains how to use the Pro-Log 7842 Z80A/RS-422 Multifunction CPU Card. It is structured to reflect the answers to basic questions you might ask yourself about the 7842. We welcome your suggestions on how we can improve our instructions.

The 7842 is part of the Pro-Log Series 7000 STD BUS hardware. Our products are modular, and designed and built with second-sourced parts that are industry standards. They provide industrial managers with the means of using their own people to control the design, production, and maintenance of the products that use STD BUS hardware.

Pro-Log supports its products with thorough and complete documentation. Also, we teach courses on how to design with and use microprocessors and the STD BUS products.

You may find the following Pro-Log documents useful in your work: the Microprocessor User's Guide and the Series 7000 STD BUS Technical Manual. If you would like a copy of these documents, please write to us on your company letterhead.

CONTENTS

	Page
Foreword	ii
Figures	v
References/Bibliography	vii
Section 1. Purpose and Main Features	1-1
Purpose	1-1
Features	1-1
Functional Description	1-3
RS-422 Interface Standard	1-4
Section 2. Installation and Specifications	2-1
Introduction	2-1
Installation	2-1
Card Layout and Jumpers	2-2
Cabling	2-2
Memory	2-6
Selecting Memory Mapping	2-7
Configuring the Memory Sockets	2-7
Disabling Unused Sockets	2-9
MEMEX	2-10
Segment	2-10
Memory Decoder PROM	2-11
I/O Mapping	2-14
I/O Decoder	2-14
IOEXP	2-16
Counter/Timer	2-16
Serial Channels	2-17
RS-232-C Interface	2-17
RS-422 Interface	2-18
Priority Chain	2-18
Specifications	2-19
Mechanical Specifications	2-24
Reset	2-29
Section 3. Operation and Programming	3-1
Introduction	3-1
Memory Expansion	3-1
Serial Communication Channels	3-1
SIO Internal Structure	3-1
SIO Receive and Transmit Registers	3-5
SIO Control Registers	3-6
SIO Status Registers	3-18

	Page
SIO Operation	3-25
Initialization	3-25
Baud Rate	3-25
General Operation	3-25
Interrupts	3-25
RS-232-C Handshaking	3-30
Block Transfers	3-31
Counter/Timer	3-31
CTC Counter Mode	3-32
CTC Timer Mode	3-34
CTC Control Registers	3-35
Load Register	3-36
Current Count Register	3-37
Vector Register	3-37
CTC Operation	3-37
Interrupts	3-37
Cascading	3-38
Output Pulses	3-38
Software Count/Trigger	3-38
CTC Timing	3-38
Section 4. Operating Software	**4-1**
Introduction	4-1
Memory Addresses	4-1
I/O Port Addresses	4-1
Description	4-1
Subroutine Specifications	4-5
(INIT COMM)	4-5
(INIT INTER)	4-5
(INIT CTC 0)	4-6
(INIT CTC 1)	4-6
(INIT SIO)	4-7
(INIT SIO CH)	4-7
(XMT CH A ASYN)	4-8
(XMT CH B SDLC)	4-9
(RCV CH A ASYN)	4-10
(RCV CH B SDLC)	4-10
(SPCL RCVA ASYN)	4-11
(SPCL RCVB SDLC)	4-11
Subroutine Flowcharts	4-14
Program Listings	4-19
Section 5. Maintenance	**5-1**
Reference Drawings	5-1
Return for Repair/Warranty	5-6

FIGURES

Figure		Page
1-1	7842 Z80A/RS-422 Multifunction CPU Card	1-2
1-2	Block Diagram	1-3
1-3	Data Signaling Rate versus Cable Length for Balanced Interface	1-5
1-4	Examples of Various Formats	1-6
2-1	Installation	2-1
2-2	Cables for Use with 7842	2-2
2-3	RS-232-C Serial Port D Connector Convertor	2-3
2-4	Connector Part Numbers	2-3
2-5	Physical Layout and Jumper Options	2-4
2-6	Memory Devices	2-6
2-7	Memory Mapping Options	2-8
2-8	Memory Socket Option Jumpers	2-9
2-9	Decoder PROM Part Numbers and Programming Equipment	2-11
2-10	Memory Decoder PROM Programming Information	2-12
2-11	Memory Decoder PROM Example, Option 0	2-13
2-12	I/O Port Functions	2-14
2-13	I/O Decoder PROM Data	2-15
2-14	Functions of W9 Jumpers	2-16
2-15	W9 Jumper Configuration	2-16
2-16	RS-232-C Serial Ports DCE/DTE Jumpers	2-17
2-17	Opcode Fetch and Refresh Timing	2-19
2-18	Memory Read Timing Other Than Opcode Fetch	2-20
2-19	Memory Write Timing	2-20
2-20	Output Timing	2-21
2-21	Input Timing	2-21
2-22	Bus Request Timing	2-22
2-23	Interrupt Request Timing	2-22
2-24	Nonmaskable Interrupt Request Timing	2-23
2-25	Wait State Timing	2-23
2-26	Electrical and Environmental Specifications	2-24
2-27	RS-422 Switching Characteristics	2-25
2-28	Channel A RS-232-C Switching Characteristics	2-25
2-29	Segment Connector J1 Pin List	2-26
2-30	Counter/Timer Connector J2 Pin List	2-26
2-31	RS-232-C Serial Connector J3 Pin List	2-27
2-32	RS-422 Serial Connector J4, J5 Pin List	2-27
2-33	STD BUS Connector Pin List	2-28
3-1	SIO Register Structure	3-2
3-2	Write Register Bit Maps	3-3
3-3	Read Register Bit Maps	3-4
3-4	SIO Control Register Functions	3-6
3-5	Write Register 0 Bit Map	3-7
3-6	Write Register 1 Bit Map	3-9
3-7	Write Register 2 Bit Map	3-11
3-8	Write Register 3 Bit Map	3-12

Figure		Page
3-9	Write Register 4 Bit Map	3-14
3-10	Write Register 5 Bit Map	3-15
3-11	Write Register 6 Bit Map	3-17
3-12	Write Register 7 Bit Map	3-18
3-13	Status Register Functions	3-19
3-14	Read Register 0 Bit Map	3-19
3-15	Read Register 1 Bit Map	3-22
3-16	Read Register 2 Bit Map	3-24
3-17	Baud Rate Selection	3-26
3-18	Interrupt Vectors and Priority	3-29
3-19	SIO RS-232-C Handshake Capability	3-31
3-20	CTC Channels per Function	3-32
3-21	CTC Internal Structure	3-33
3-22	CTC Control Registers	3-35
4-1	Subroutine Types	4-2
4-2	Main Subroutines Use Requirements	4-4
4-3	1K Program Memory Map	4-12
4-4	256 Byte RAM Memory Map	4-13
4-5	(INIT COMM) Flowchart	4-14
4-6	(INIT CTC 0) Flowchart	4-14
4-7	(INIT CTC 1) Flowchart	4-14
4-8	(INIT INTER) Flowchart	4-14
4-9	(INIT SIO) Flowchart	4-14
4-10	(INIT SIO CH) Flowchart	4-15
4-11	(XMT CHA ASYN) Flowchart	4-15
4-12	(XMT CHB SDLC) Flowchart	4-16
4-13	(XMT CHAR) Flowchart	4-16
4-14	(END XMT?) Flowchart	4-16
4-15	(XMT BLK B) Flowchart	4-16
4-16	(RCV CHA ASYN) Flowchart	4-17
4-17	(RCV CHB SDLC) Flowchart	4-17
4-18	(SPCL RCV A ASYN) Flowchart	4-18
4-19	(SPCL RCV B SDLC) Flowchart	4-18
5-1	Schematic (For Reference Only)	5-2
5-2	Assembly Drawing (For Reference Only)	5-6

REFERENCES/BIBLIOGRAPHY

Mostek MK3884/MK3885 Serial I/O Controller Technical Manual (MK78583).

Mostek 1979 Microcomputer Data Book. Mostek Corporation, 1979

Mostek 1982/1983 Microelectronic Data Book. Mostek Corporation, 1982

1982/83 Data Book. Zilog, Inc., 1982.

Microprocessor Applications Reference Book, Volume 1. Zilog, Inc., 1981.

EIA Standard* RS-232-C.

EIA Standard* RS-422-A.

EIA Standard* RS-485.

*NOTE: To obtain copies of EIA Standards, write to:

Electronic Industries Association
Engineering Department
2001 Eye Street, N.W.
Washington, D.C. 20006

PURPOSE

The 7842 is a Z80A multifunction processor card which provides processor, memory, counter/timer, and input/output capabilities on a single STD BUS card. The 7842 combines a 4 MHz Z80A processor with a Z80A serial input/output controller (SIO), two RS-422 channels, a 4-channel counter/timer, and four 28-pin memory sockets.

The two serial channels are fully independent. One channel provides RS-422 interface levels, and the second channel can be configured to either RS-232-C or RS-422 interface levels. The 7842 operates in asynchronous, byte synchronous (BISYNC), and bit synchronous (SDLC, HDLC, CCITT-X.25) modes. The RS-422 interface allows 5-volt only card operation. The 7842 also meets the RS-485 multidrop standard, which enables 32 driver/receiver pairs to be placed on a single line. The counter/timer has four independent timer channels. These timer channels can be used as external event counters or as prioritized interrupt inputs with vectors. Two of the channels may be used to generate baud rates for the SIO ports. Two vectored general interrupts or vectored interrupts for the counter/timer channels are available, plus vectored interrupts for the serial communications channels.

FEATURES

- Three versions: 4.0 MHz (7842-0), 3.6864 MHz (7842-1), and 2.5 MHz (7842-2) operation
- Two independent full duplex serial communications channels
- RS-422 or multidrop RS-422 (RS-485) interfaces, with one channel configurable as RS-232-C
- 5V only operation with both channels configured as RS-422
- Onboard serial input/output controller
- Asynchronous, BISYNC, SDLC, HDLC, CCITT-X.25 protocols
- RS-422 synchronous data signaling rates up to 500K baud (with a 4.0 MHz clock)
- RS-422 asynchronous data signaling rates up to 62.5K baud (with a 4.0 MHz clock)
- RS-232-C synchronous and asynchronous data signaling rates up to 19.2K baud

- Transmission distance of up to 4,000 feet
- Fully buffered Z80A processor
- Capacity for up to 128K EPROM or 32K RAM onboard
- Bootstrap EPROM capability
- Flexible memory mapping
- Z80A block I/O transfer capability
- Four independent 8-bit counter/timer channels
- Three counter/timer channels to be used as external event counters or prioritized vectored interrupts

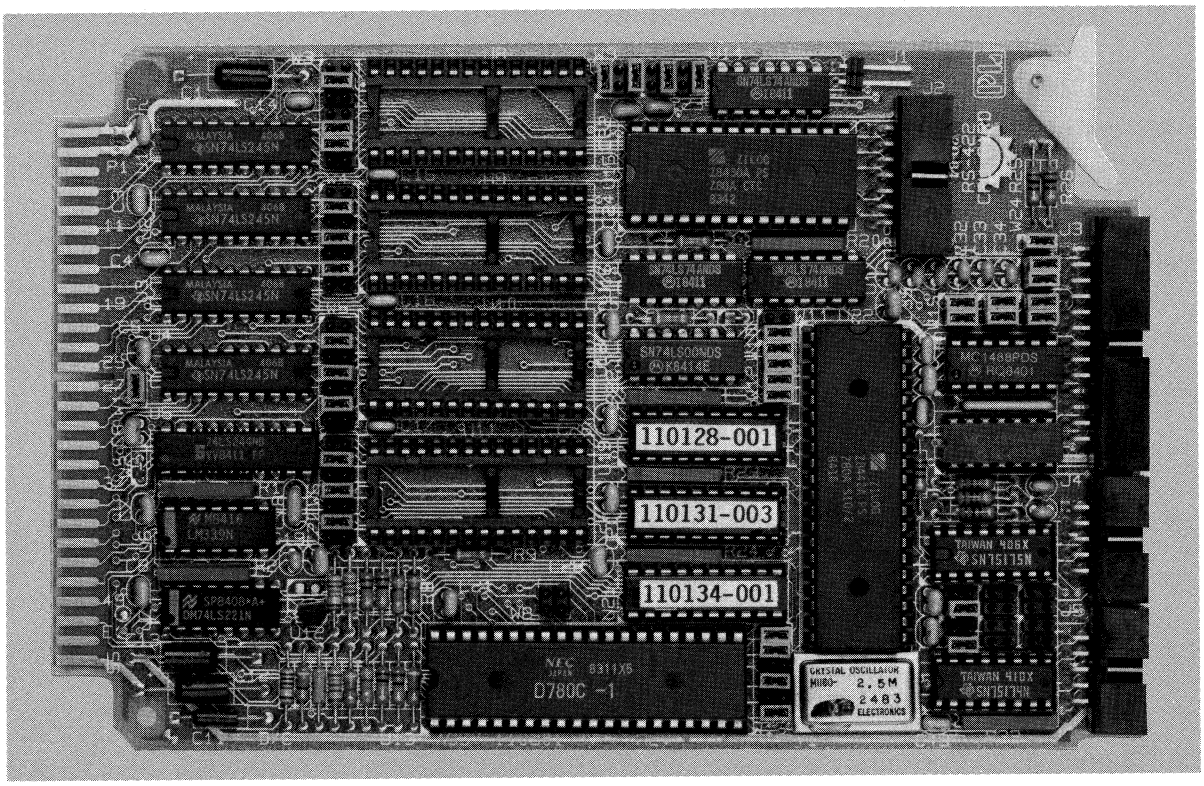


Figure 1-1. 7842 Z80A/RS-422 Multifunction CPU Card

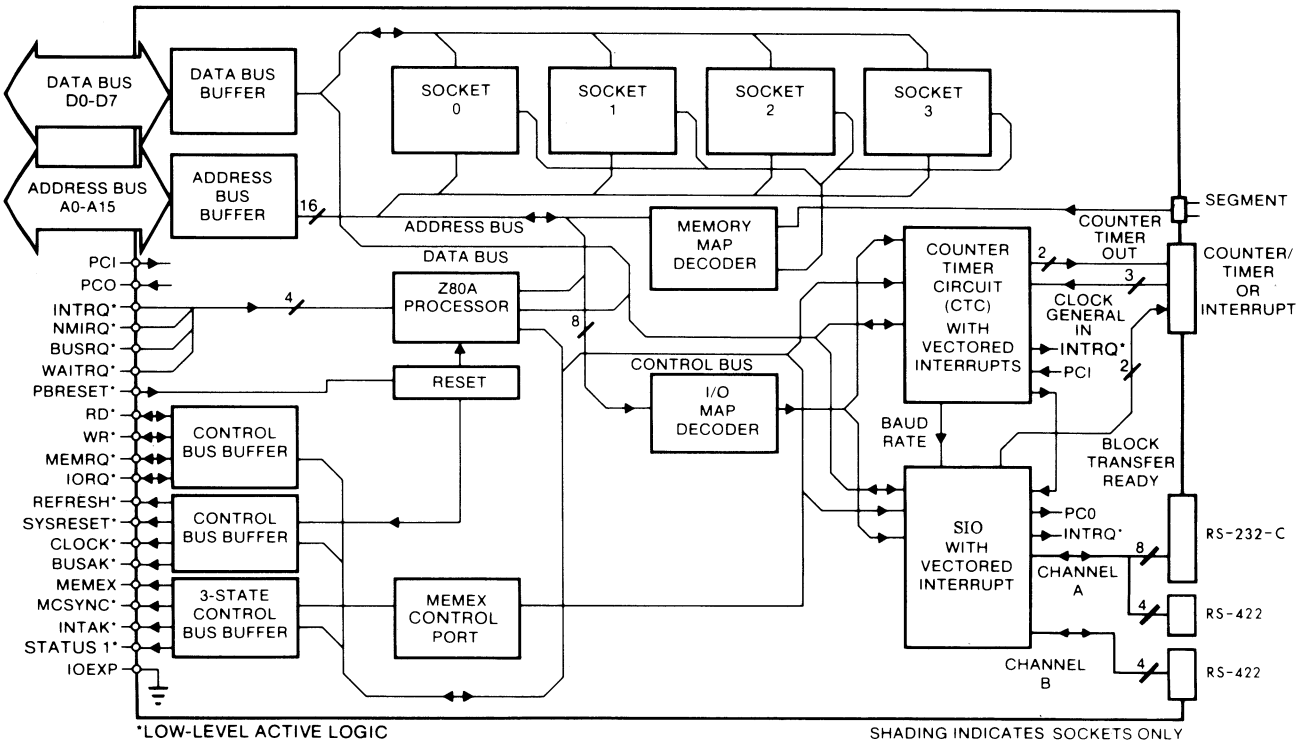


Figure 1-2. Block Diagram (7842)

FUNCTIONAL DESCRIPTION

The functional description defines each block in the 7842 block diagram (figure 1-2). The block diagram and the functional description are designed to familiarize the user with terms used throughout the manual.

Data Bus Buffer — an octal bus transceiver designed to provide buffering and direction control of the data bus.

Address Bus Buffer — two octal bus transceivers designed to provide buffering and direction control of the address bus.

Control Bus Buffer -- provides the control signal interface between the STD BUS and the card. The RD*, WR*, MEMRQ*, and IORQ* buffers are bidirectional for DMA applications.

Memory Sockets — four memory sockets which can accept bytewide RAMs or EPROMs. As shipped, memory options are for three 4K EPROMs in sockets 0, 1, and 2, and a 2K RAM in socket 3.

Z80A Processor — system's central processing unit, operable with a 4.0 MHz (7842-0), 3.6864 MHz (7842-1), or 2.5 MHz (7842-2) clock.

Reset — controls the SYSRESET* line. A power-on reset holds SYSRESET* low until after VCC reaches 4.75 (nominal). The PBRESET* input is debounced and produces a SYSRESET pulse.

Memex Control Port — selects polarity of MEMEX at power-up and during memory bank selection.

Memory Map Decoder — selects the size of memory space to be occupied by each memory socket. The bipolar PROM provided contains seven pre-programmed and nine user programmable memory mapping schemes.

I/O Map Decoder — selects the port addresses for the CTC, the SIO, and MEMEX. The bipolar PROM provided contains one pre-programmed and one user programmable I/O mapping scheme.

CTC - the counter/timer circuit can be used for timing, counting, generating vectored interrupts, and generating baud rate clocks.

SIO — the serial input/output controller directs data communications across the serial interface channels. This peripheral operates in asynchronous, byte synchronous, and bit synchronous modes. It can also generate vectored interrupts.

Segment Select Connector — selects a bank of memory, either onboard or off-board.

Counter/Timer Connector — provides user access to counter/timer, wait/ready, and vectored interrupt functions.

SIO Connectors — provides user interface to the SIO. A 26-pin RS-232-C connector and two 10-pin RS-422 connectors are provided.

RS-422 INTERFACE STANDARD

The 7842 combines the Z80A microprocessor with intelligent peripheral devices to provide multiple functions basic to a microprocessor-based system (see block diagram, figure 1-2, page 1-2.) The serial input/output controller (SIO) has two completely independent full duplex communications channels which can operate with independent baud rates. Channel A can be configured for RS-232-C or RS-422 operation, with channel B dedicated to RS-422 interface levels. The RS-422 interface is used in serial data communications applications requiring long cable lengths, high noise immunity, or high data rates. See References/Bibilography, page vii, for information on obtaining a copy of the EIA RS-422-A standard. Using a 24 AWG twisted pair line, the 7842 CPU card can transmit and receive data a maximum distance of 4,000 feet (1.2 km) at synchronous data rates as high as 90K baud.

The maximum permissible length of cable separating the transmitter and receiver is a function of data signaling rate and is influenced by the allowable signal distortion, the amount of longitudinally coupled noise, the difference in ground potential introduced between the transmitter and receiver grounds, and cable balance. Increasing the cable distance between the transmitter and receiver interfaces increases exposure to common mode noise, signal distortion, and the effects of cable imbalance.

The curve of cable length versus data signaling rate given in figure 1-3, from the EIA RS-422 standard, may be used as a conservative guide. This curve is based upon empirical data using a 24 AWG, copper conductor, twisted-pair telephone cable with a shunt capacitance of 52.5 pF/meter (16 pF/foot) terminated in a 100 ohm resistive load. The cable length restriction shown by the curve is based upon assumed load signal quality requirements of:

a. Signal rise and fall time equal to, or less than, one-half unit interval at the applicable modulation rate.

b. A maximum voltage loss between generator and load of 6dB (voltage).

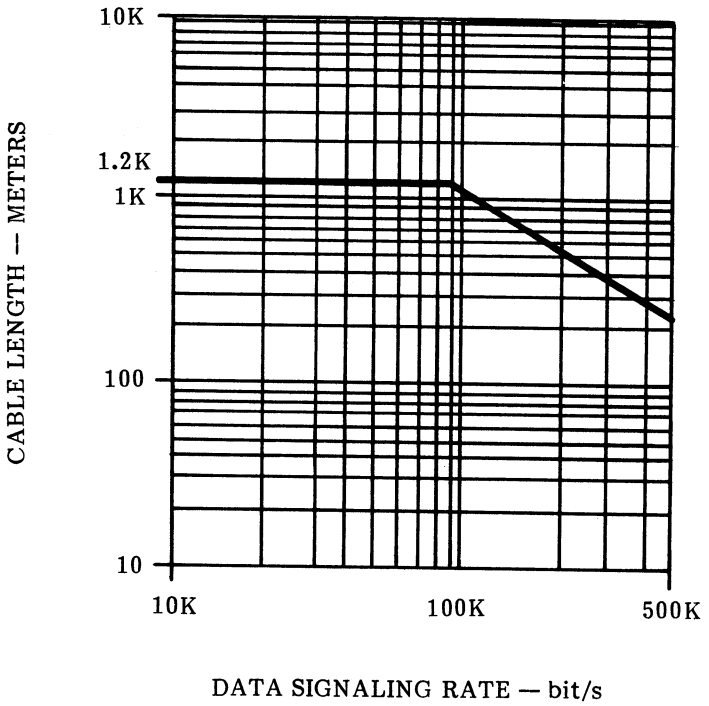
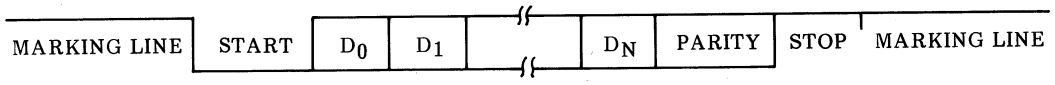


Figure 1-3. Data Signaling Rate vs Cable Length for Balanced Interface

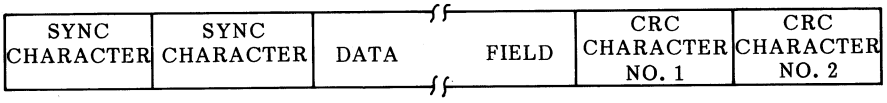
The 7842/RS-422 serial communications connector provides a minimum number of wires for adequate synchronous data transmission. Data, clock, and ground signals are provided. All handshaking is accomplished in software. The pinout for the RS-422 connector was selected to eliminate the need for DTE/DCE channel configuration by making it doubly symmetrical. See Pro-Log's Application Note #140 for more information.

The SIO is capable of transmitting and receiving data in various formats: asynchronous, byte synchronous, and bit synchronous. Examples of each are shown in figure 1-4. The communications format of each I/O channel is determined by the contents of the software programmable SIO control registers.

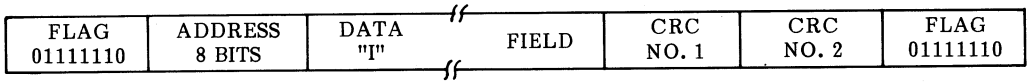
Jumpers are provided to configure the RS-422 interfaces to RS-485 multidrop interfaces, allowing 32 driver/receiver pairs to be placed on a single line. The interface drivers have separate enables and thermal shutdown capability to handle contention on the line.



ASYNCHRONOUS FORMAT



BISYNC FORMAT



SDLC/HDLC FORMAT

Figure 1-4. Examples of Various Formats

INTRODUCTION

This section covers all aspects of the 7842 card which may need to be reviewed before operation. This includes optional functions selected by shorting plugs, cables to be used, and installation. Complete specifications for the 7842 are at the end of this section.

INSTALLATION

> **CAUTION**
>
> To prevent possible damage to your STD BUS system, make sure power is **off** before inserting a card into the card rack or before removing a card from the card rack.

The 7842 can operate independently or as part of an STD BUS system. If used as part of an STD BUS system, it should be installed in an STD BUS backplane, card rack, or Bus Box with the card ejector toward the top as shown in figure 2-1. It can be installed in any slot in the rack. If access is needed to the card for testing or servicing, a card extender may be used such as Pro-Log's model 7901.

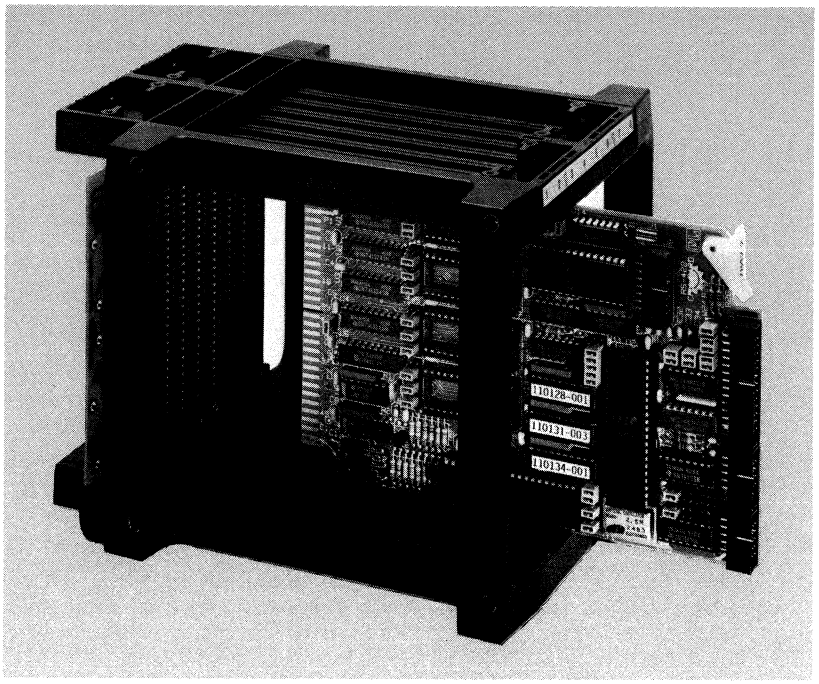


Figure 2-1. Installation (7842)

CARD LAYOUT AND JUMPERS

Several optional functions on the 7842 are selected by jumpers. Most of the jumpers are wirewrap posts which can be shorted by shorting plug or wirewrapping. The part number for additional shorting plugs is Elco Corp. part number 00 8261 02 42 00 870. Two of the jumpers are soldered-in wire jumpers. These can be disconnected by cutting them. Be sure to cut them in two places to prevent reshorting. The wire jumpers can be replaced with a pair of wirewrap posts (Elco Corp. part number 00 8261 02 32 00 852). When removing the wire jumpers, cut them and remove one half at a time to prevent circuit trace damage.

CABLING

Figure 2-2 is a list of cables which may be used with the 7842. Figure 2-3 shows the pinouts for connectors of the RS-232-C transition cable (parallel pin to the D Connector).

7806 CONNECTOR	PRO-LOG CABLE	DESCRIPTION
J3 RS-232-C Connector	RC701-3 or RC702-3	3 feet, terminates in 25-pin female (RC701-3) or male (RC702-3) D connector with RS-232-C pinout.
J2 CTC and W/Ready Lines Connector	RC704-1 or RC704-2	2-pin connector (same on both ends) with 1 foot (RC704-1) or 2 feet (RC704-2) of twisted pair cable, can mate to interrupt output connectors of other cards or can mate to 7605 TTL I/O card for segment.
J1 Segment Connector		

Figure 2-2. Cables for Use with 7842

PIN # OF SERIAL CONNECTOR J1 OR J2	SIGNAL NAME	RS-232-C PIN
1	N/C	1
2	N/C	14
3	TD*	2
4	RCLK (DB)	15
5	RD*	3
6	N/C	16
7	RTS	4
8	RCLK (DD)	17
9	CTS	5
10	N/C	18
11	DSR	6
12	N/C	19
13	GND	7
14	DTR	20
15	N/C	8
16	N/C	21
17	N/C	9
18		22
19	N/C	10
20	N/C	23
21	N/C	11
22	TCLK (DA)	24
23	N/C	12
24	N/C	25
25	N/C	13
26	N/C	N/C

Figure 2-3. RS-232-C Serial Port D Connector Convertor

There are several methods for connecting the RS-422 interface (see Pro-Log Application Note #104). Figure 2-4 lists female transition connectors which mate with the RS-422 connector on the 7842 card.

MANUFACTURER	PART NUMBER
Ansley	609-1030-1
Berg	65847-004-A
Burndy	FRS 10 DB-1P
Robinson-Nugent	1DS-10-SR
SAE	RD6310BRSC
3M	3473-7010

Figure 2-4. Connector Part Numbers

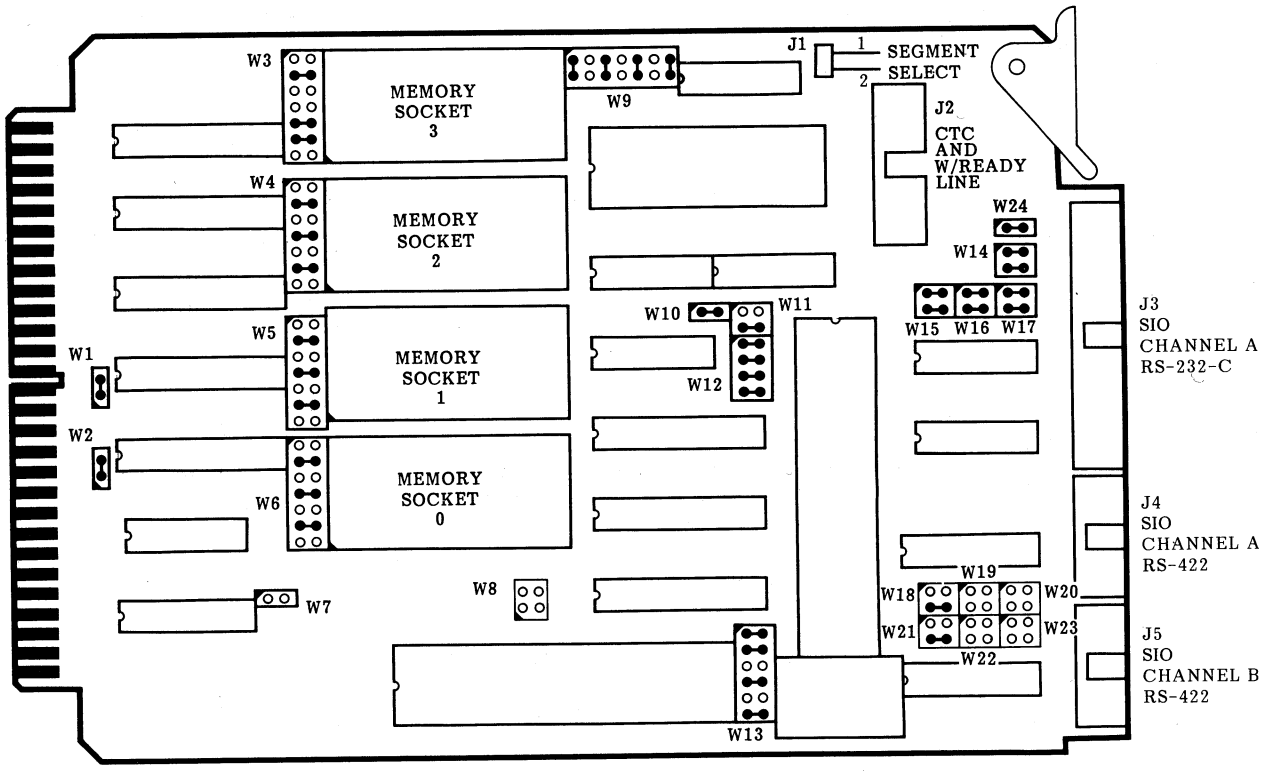


Figure 2-5. Physical Layout and Jumper Options (7842)

Jumper Group		DESCRIPTION	As Shipped
W1	1-2 none	7842 controls **MEMEX** MEMEX floating, must be controlled by another card.	1-2
W2	1-2 none	**IOEXP** connected to ground IOEXP floating, must be controlled by another card.	1-2
W3-W6		Configure **Memory Sockets** 3-0 respecively for memory size and ROM or RAM. See figure 2-8.	W3 3-4, 9-10, 11-12 W4-W6 3-4, 7-8, 11-12
W7		For Pro-Log use only.	
W8	1-2 3-4	Controls block transfers to channel A Controls block transfers to channel B	No jumpers
W9		Configures channels A and B transmit and receive baud rates. See figure 2-15.	1-2, 5-6, 9-10, 13-14
W10	1-2 None	Standard **I/O Mapping** User programmed I/O mapping	1-2
W11	1-2 3-4	**MEMEX** high at power-up and MEMEX port inverting MEMEX low at power-up and MEMEX port noninverting	3-4
W12	1-2 thru 7-8	Enable **Memory sockets** 0-3 respectively	1-2, 3-4, 5-6, 7-8
W13		Affects **Memory Mapping** and response of onboard memory to MEMEX and Segment. See figure 2-7.	1-2, 3-4 7-8, 11-12
W14-17, 24		Affects RS-232-C interface. See figure 2-16.	1-2, 3-4
W18, 21	1-2 3-4	Selects RS-232-C interface for channel A Selects RS-422 interface for channel A	3-4
W19, 20 22, 23	1-2 3-4	Connects transmit (+) and receive (+) lines for multidrop applications. Connects transmit (-) and receive (-) lines for multidrop applications	No jumpers

Figure 2-5. (Continued)

MEMORY

The 7842 has four sockets which accept RAM or ROM memory parts in a variety of sizes. Memory parts used must have an access time no greater than 250 ns for 4 MHz and 3.6864 MHz versions (7842-0 and 7842-1), or 450 ns for 2.5 MHz versions (7842-2). 2K, 4K, 8K, 16K, and 32K EPROMs and 2K and 8K RAMs can be used on the 7842 card. Figure 2-6 lists some recommended memory devices. There are eight jumpers affecting memory mapping. Memory decoding on the 7842 is controlled by a bipolar PROM. The PROM contains 16 optional memory mapping schemes, 7 of which have been programmed. The rest are blank and may be programmed by the user. The active option is selected by jumper. The signals controlling memory include the STD BUS memory Expansion line (MEMEX), which allows two 64K memory banks in a system, and a Segment line. Segment is an external signal decoded by this and most Pro-Log memory cards which allows a virtually unlimited number of memory banks in a system.

	MEMORY SIZE	MANUFACTURER PART NUMBER	ACCESS TIME (ns)
EPROM	2K[1]	Intel 2716 Mostek MK2716	350-650 300
	4K	Intel 2732A NEC uPD2732	200, 250 no spec
	8K	Intel 2764 Mostek MK2764	200-250 450
	16K	Intel 27128 AMD 27128	250 150-450
	32K	Intel 27256 AMD 27256	250 170-450
RAM	2K	Mitsubishi M58725 OKI MSM 5128	200 150-200-250
	8K	Hitachi HM6264LP-15 Toshiba TC5565PL-15	150 150

Figure 2-6. Memory Devices[2]

[1] Can only be used with the 7842-2 2.5 MHz version.
[2] 4K and 16K RAMs do not exist as of this writing. Pro-Log believes the 7842 will support these chips when they become available.

Selecting Memory Mapping

Jumper W13 selects memory mapping (figures 2-5 and 2-7, pages 2-4 and 2-9). The memory decoder PROM has seven preprogrammed options. The rest are blank and may be programmed by the user (see Memory Decoder PROM, page 2-11). MEMEX and Segment are also included in memory decoding (see MEMEX and SEGMENT, page 2-10). The memory option selects only the size of memory space to be occupied by each socket. Whether a socket is enabled or not, and whether it is configured for ROM or RAM is determined by the jumpers described in the following headings.

Configuring the Memory Sockets

After a memory option has been selected, the memory sockets must be configured accordingly. They must be configured for the correct size and may be configured for either ROM or RAM. Figure 2-8, page 2-9, shows how the jumpers should be arranged for each memory type. Jumpers W3-W6 are used for this purpose and correspond to sockets 3-0 respectively. Figure 2-5, page 2-4, shows the locations of the sockets and jumpers.

OPTION #	A8[1]	A7[2]	A6	A5	SOCKET 0	SOCKET 1	SOCKET 2	SOCKET 3
0	LO	LO	LO	LO	4K 0000-0FFF	4K 1000-1FFF	4K 2000-2FFF	2K 3000-37FF
1	LO	LO	LO	HI	8K 0000-1FFF	8K 2000-3FFF	8K 4000-5FFF	8K 6000-7FFF
2	LO	LO	HI	LO	16K 0000-3FFF	16K 4000-7FFF	8K 8000-9FFF	8K A000-BFFF
3	LO	LO	HI	HI	32K 0000-7FFF	8K 8000-9FFF	8K A000-BFFF	8K C000-DFFF
4	LO	HI	LO	LO	4K 0000-0FFF	4K 1000-1FFF	4K 2000-2FFF	4K 3000-3FFF
5	LO	HI	LO	HI	16K 0000-3FFF	12K 4000-6FFF	2K 7000-77FF	2K 7800-7FFF
6	LO	HI	HI	LO	2K 0000-07FF	2K 0800-0FFF	2K 1000-17FF	2K 1800-1FFF
7	LO	HI	HI	HI				
8	HI	LO	LO	LO				
9	HI	LO	LO	HI		UNPROGRAMMED, MAY BE PROGRAMMED BY THE USER		
10	HI	LO	HI	LO				
11	HI	LO	HI	HI				
12	HI	HI	LO	LO				
13	HI	HI	LO	HI				
14	HI	HI	HI	LO				
15	HI	HI	HI	HI				

NOTE: With jumper W13 configured as shipped and with MEMEX low, as it is after power-up or reset, option #0 is selected.
[1]May be connected to MEMEX
[2]May be connected to Segment

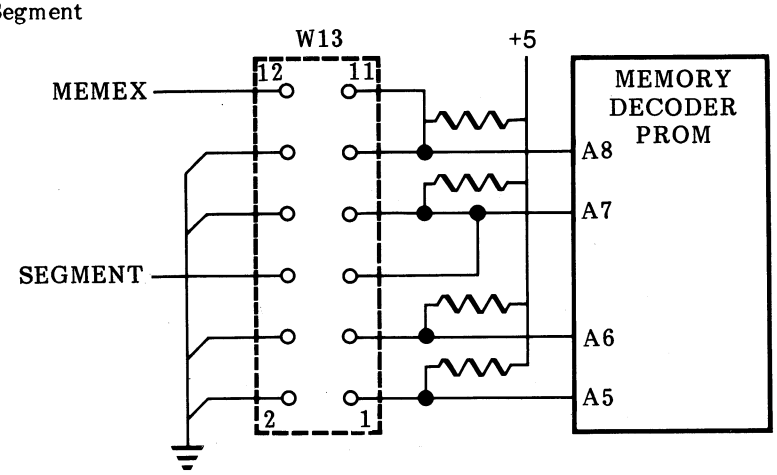
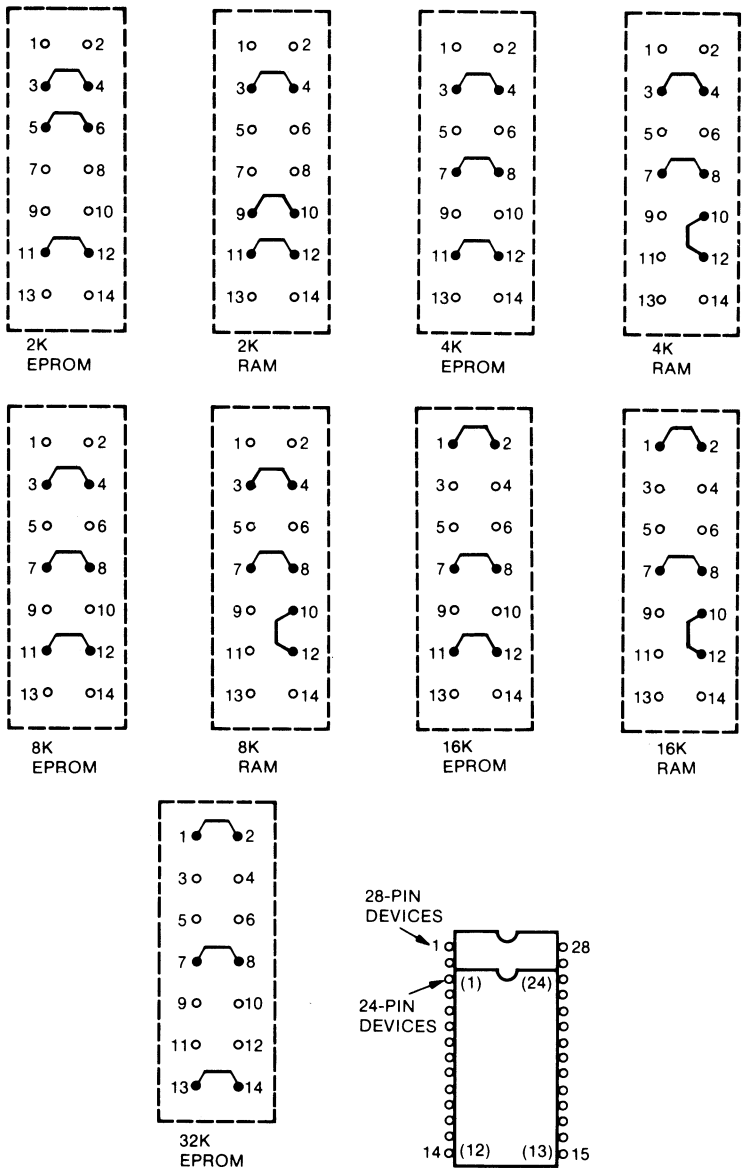


Figure 2-7. Memory Mapping Options (7842)



NOTES: 1. Jumper groups W3-W6 correspond to sockets 3-0 respectively.

2. 4K and 16K RAMs do not exist as of this writing. Pro-Log believes the 7842 will support these chips when they become available.

Figure 2-8. Memory Socket Option Jumpers

Disabling Unused Sockets

Unused memory sockets can be disabled by removing jumpers from W12. Jumpers 1-2 through 7-8 correspond to sockets 0-3 respectively. Removing a jumper causes the processor card to expect memory to be off-card over the address range of the socket, thus causing the socket to effectively disappear from the memory map. If a socket does not occupy any memory space according to the memory decoder PROM option, the socket does not need to be disabled. Sockets which are disabled should be left empty or data bus contention may occur.

MEMEX

MEMEX is the STD BUS memory expansion line. It selects one of two banks of memory with up to 64K of memory per bank. It can be used to expand memory to to control a bootstrap PROM. In bootstrapping, the bootstrap PROM is enabled initially. After bootstrapping, the MEMEX line is complemented, disabling the PROM and enabling a block of RAM in its place. The 7842 controls MEMEX through an onboard output port.

There are three jumpers affecting MEMEX (figure 2-5, page 2-4). Jumpers W1 and W11 affect the 7842 control of MEMEX. Jumper W13 pins 11-12 and the memory decoder PROM affect the 7842 response of the onboard memory to MEMEX (figure 2-7, page 2-8).

With jumper W12 pins 11-12 removed, the onboard memory does not respond to MEMEX. With the jumper in place, the onboard memory responds to MEMEX according to the way the memory decoder PROM is programmed. Note that jumper W12 pins 9-10 should not be shorted if pins 11-12 are shorted.

When MEMEX changes state, a different memory decoder PROM option is selected. For instance, if option 0 is selected when MEMEX is low, option 8 will be selected when MEMEX is high.

The two banks MEMEX selects may be either one bank onboard and one bank offboard or both banks onboard. If bootstrapping is used, both banks may be onboard. The bootstrap EPROM may be in sockets 0 and 1, and the RAM replacement in sockets 2 and 3.

Segment

The Segment signal is similar to MEMEX. It is not an STD BUS signal. Segment would typically be generated by a port on an I/O card and connected to a memory card or the 7842 by a cable. Most Pro-Log memory cards decode Segment.

Several Segment lines are generated by different ports or bits of a port. Only one is active at a time, allowing selection of one of any number of memory banks.

With jumper W13 pins 5-6 removed, onboard memory does not respond to Segment (figures 2-5 and 2-7, pages 2-4 and 2-8). With 5-6 installed, Segment selects one of two memory decoder PROM options. For instance, if option 0 is selected with Segment low, option 4 is selected with Segment high. Note that jumper W13 pins 5-6 and 7-8 must not be connected at the same time.

Like MEMEX, the effect of Segment on onboard memory is a function of the memory decoder PROM.

Memory Decoder PROM

The memory decoder PROM is a 512 by 8 bipolar PROM. Figure 2-9 is a list of parts which can be used as a decoder PROM and the Pro-Log PROM programming equipment to program them. The same PROM and programming equipment are used to program the I/O decoder.

PROM PART #	Pro-Log PROM Programming Equipment*		
	PERSONALITY MODULE	PINOUT ADAPTER	CONFIG-URATOR
MMI 6348-1	PM9037	PA20-1	512x8(H)
Harris 7648-5	PM9039A	PA20-1	512x8(H)
TI TBP 285A42	PM9067	PA20-1	512x8(H)

*The equipment listed must be used in conjunction with any Pro-Log Series 90 PROM Programmer.

Figure 2-9. Decoder PROM Part Numbers and Programming Equipment (7842)

The decoder PROM contains 16 memory mapping options. Seven of these have been preprogrammed (figure 2-7, page 2-8); the rest are blank and may be programmed by the user. Each option occupies 32 bytes in the PROM and each byte represents a 2K byte section of memory. Depending on the data programmed into each location, the decoder selects one of the onboard memory sockets or offcard memory over the 2K address space represented by that byte.

Figure 2-10 shows the 32 byte section of the PROM representing each option, the address range that each byte within an option affects, and the data byte to be programmed into each byte to select an onboard socket or offcard memory. Figure 2-11 shows, as an example, the program already in the decoder PROM for option 0. If a socket is to occupy more than 2K of memory space, simply repeat the appropriate data byte as many times as necessary. Note that in the example, sockets 0, 1, and 2 each occupy 4K of memory space.

If MEMEX or Segment are to be used, two options need to be programmed: one for when the signal is low and the other when it is high. Refer to figure 2-7, page 2-8, and the MEMEX and Segment headings in this section.

(Option Addresses)

DECODER PROM ADDRESS	OPTION #
000-01F	0
020-03F	1
040-05F	2
060-07F	3
080-09F	4
0A0-0BF	5
0C0-0DF	6
0E0-0FF	7
100-11F	8
120-13F	9
140-15F	10
160-17F	11
180-19F	12
1A0-1BF	13
1C0-1DF	14
1E0-1FF	15

(Program Code)

DATA BYTE	MEMORY ENABLED
FF	OFF CARD
EE	SOCKET 0
DD	SOCKET 1
BB	SOCKET 2
77	SOCKET 3

BYTE # WITHIN AN OPTION		AFFECTED ADDRESS RANGE
DEC	HEX	
1	00	0000-07FF
2	01	0800-0FFF
3	02	1000-17FF
4	03	1800-1FFF
5	04	2000-27FF
6	05	2800-2FFF
7	06	3000-37FF
8	07	3800-3FFF
9	08	4000-47FF
10	09	4800-4FFF
11	0A	5000-57FF
12	0B	5800-5FFF
13	0C	6000-67FF
14	0D	6800-6FFF
15	0E	7000-77FF
16	0F	7800-7FFF
17	10	8000-87FF
18	11	8800-8FFF
19	12	9000-97FF
20	13	9800-9FFF
21	14	A000-A7FF
22	15	A800-AFFF
23	16	B000-B7FF
24	17	B800-BFFF
25	18	C000-C7FF
26	19	C800-CFFF
27	1A	D000-D7FF
28	1B	D800-DFFF
29	1C	E000-E7FF
30	1D	E800-EFFF
31	1E	F000-F7FF
32	1F	F800-FFFF

Figure 2-10. Memory Decoder PROM Programming Information (7842)

PRO-LOG CORPORATION

PROGRAM ASSEMBLY FORM

HEXADECIMAL PAGE ADR	HEXADECIMAL LINE ADR	HEXADECIMAL INSTR.	MNEMONIC LABEL	MNEMONIC INSTR.	MNEMONIC MODIFIER	TITLE MEMORY DECODING DATE / COMMENTS
0	00	EE				ENABLE ROM/RAM 0
0	01	EE				
0	02	DD				ENABLE ROM/RAM 1
0	03	DD				
0	04	BB				ENABLE ROM/RAM 2
0	05	BB				
0	06	77				ENABLE ROM/RAM 3
0	07	FF				EXTERNAL MEMORY
0	08					
0	09					
0	0A					
0	0B					
0	0C					
0	0D					
0	0E					
0	0F					
0	10					
0	11					
0	12					
0	13					
0	14					
0	15					
0	16					
0	17					
0	18					
0	19					
0	1A					
0	1B					
0	1C					
0	1D					
0	1E					
0	1F	FF				

100001 2/77

Figure 2-11. Memory Decoder PROM Example, Option 0 (7842)

I/O MAPPING

Two jumpers affect I/O mapping. I/O mapping is controlled by a bipolar PROM in much the same way as memory mapping. The I/O decoder PROM has one mapping option preprogrammed into it and one option which is blank. The 7842 occupies nine I/O addresses: four for the CTC, four for the SIO, and one for MEMEX (see figure 2-12). By programming the second option of the memory decoder PROM, the user may select almost any I/O mapping plan.

I/O PORT ADDR	OUTPUT FUNCTION	INPUT FUNCTION
F0	CTC CHANNEL 0 Control, Vector, and Load Registers	CTC CHANNEL 0 Current Count Register
F1	CTC CHANNEL 1 Control and Load Registers	CTC CHANNEL 1 Current Count Register
F2	CTC CHANNEL 2 Control and Load Registers	CTC CHANNEL 2 Current Count Register
F3	CTC CHANNEL 3 Control and Load Registers	CTC CHANNEL 3 Current Count Register
F4	SIO CHANNEL A XMT Data	SIO CHANNEL A RCV Data
F5	SIO CHANNEL A Control Registers	SIO CHANNEL A Status Registers
F6	SIO CHANNEL B XMT Data	SIO CHANNEL B RCV Data
F7	SIO CHANNEL B Control Registers	SIO CHANNEL B Status Registers
FE	MEMEX CONTROL	NOT USED

Figure 2-12. I/O Port Functions (7842)

I/O Decoder

Figure 2-9, page 2-11, is a list of the type of PROMs which can be used as I/O decoder PROMs, and the Pro-Log PROM programming equipment to program them. I/O decoder PROM program development differs from that of the memory decoder PROM.

The I/O PROM has 512 bytes. The first 256 represent the first option and the last 256 represent the second option. Every byte represents one I/O address. Program each location with the data (figure 2-13) to select the desired function for the corresponding I/O address.

The SIO and CTC each occupy four I/O addresses and should have four addresses programmed for their use. Since they each decode address lines A0 and A1, they should occupy four consecutive addresses.

The second option, programmed by the user, is activated by removing jumper W10 (figure 2-5, page 2-4).

DATA BYTE	I/O SELECTED
FF	OFF CARD
F6	CTC
F5	SIO
F3	MEMEX

Figure 2-13. I/O Decoder PROM Data (7842)

IOEXP

IOEXP is the I/O expansion line of the STD BUS. The 7842 does not control IOEXP or decode it for its own use. Most Pro-Log I/O cards do decode it and must have it low to operate. IOEXP is connected to ground on the 7842. If another card in the system controls IOEXP, jumper W2 should be removed. This leaves IOEXP floating and free to be controlled externally. The 7842 onboard I/O ports are independent of IOEXP and will appear on both the IOEXP(H) and IOEXP(L) maps. Thus, these port addresses should not be used on other cards in the system. The port addresses may be remapped (see I/O Mapping, page 2-14).

COUNTER/TIMER

The 7842 can transmit data synchronously or asynchronously. The two SIO channels can be jumpered to have the same baud rate source (generated by CTC ch 0) or different baud rate sources (CTC ch 0 and ch 1). Combinations of these features are user selected by configuration of the seven jumpers of W9 (see figure 2-15). The functions of individual jumpers of W9 are shown in figure 2-14. As shipped, CTC channel 0 provides the baud rate clock for SIO channel A and CTC channel 1 provides the baud rate clock for SIO channel B. Both SIO channels are configured for synchronous communication (the receive baud rate clocks are from an external source).

JUMPER	PINS	FUNCTION
W9	1-2	SIO channel B receives baud clock from an external source
	3-4	SIO channel B receives baud clock from the CTC
	5-6	CTC channel 1 generates baud clock for SIO channel B
	7-8	CTC channel 0 generates baud clock for both SIO channels
	9-10	System clock÷2 is input to CTC channel 1
	11-12	SIO channel A receives baud clock from CTC channel 0
	13-14	SIO channel A receives baud clock from an external source

Figure 2-14. Functions of W9 Jumpers (7842)

	SIO CHANNEL A		SIO CHANNEL B	
	ASYNCH	SYNCH	ASYNCH	SYNCH
CTC channel 0	11-12	13-14	3-4, 7-8	1-2, 7-8
CTC channel 1	NOT AVAILABLE		3-4, 5-6, 9-10	1-2, 5-6, 9-10

Figure 2-15. W9 Jumper Configuration (7842)

SERIAL CHANNELS

The 7842 has two serial communications channels: channel A is RS-422/RS-232-C interchangeable, and channel B is RS-422 only. Jumpers W18 and W21 configure channel A as RS-232 (pins 1-2) or RS-422 (pins 3-4, as shipped). If the RS-422 interface is selected for channel A, the 7842 can operate with a 5V-only power supply.

NOTE: If +12V power supplies are not provided to the card, the RS-232-C receiver (U22) should be removed from its socket. This prevents erroneous interrupts.

RS-232-C Interface

The RS-232-C interface contains four jumpers (see figure 2-16) which select whether the channel operates as data communications equipment (DCE) or data terminal equipment (DTE). When operating channel A as DCE, jumpers W14-W17 should have jumpers across pins 1-2 and 3-4. To operate channel A as DTE, turn the jumpers sideways to short pins 1-3 and 2-4. If a transmit clock is required as a DTE input, jumper W24 is installed. As shipped, the RS-232-C interface is in the DCE configuration and W24 is installed.

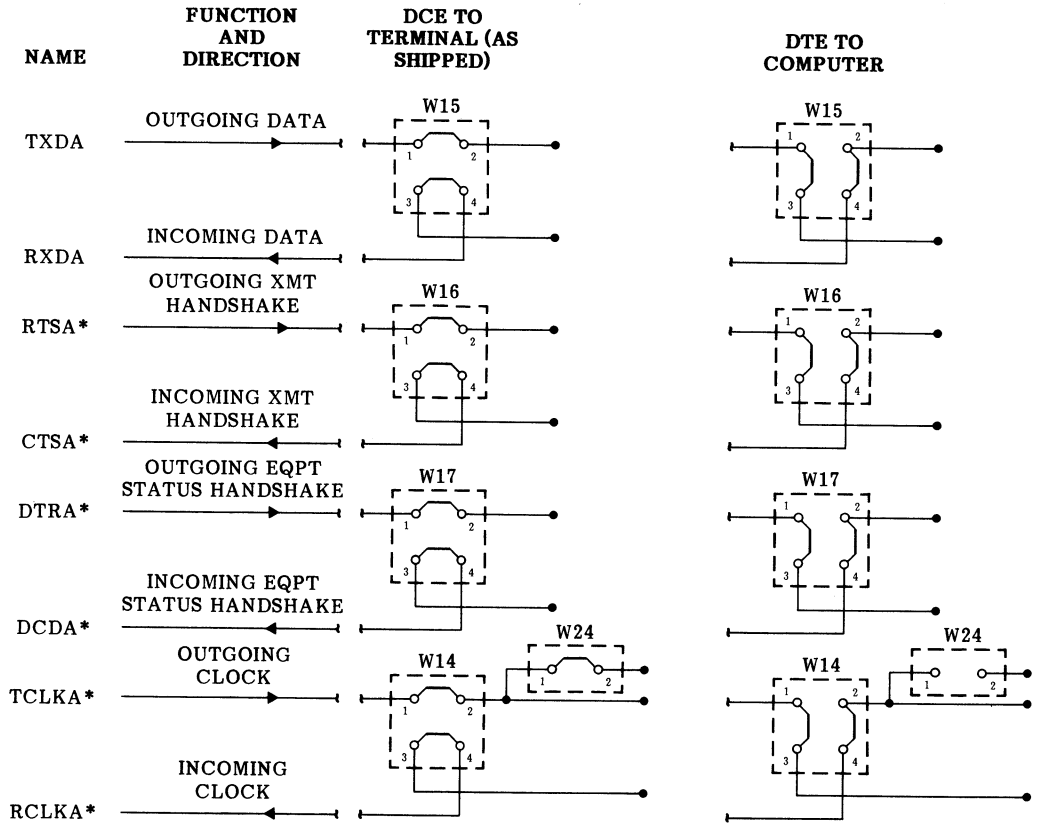


Figure 2-16. RS-232-C Serial Ports DCE/DTE Jumpers (7842)

RS-422 Interface

The pinout for the RS-422 connector eliminates the need for DTE/DCE channel configuration and preserves the differential signals on twisted pairs (see Pro-Log Application Note #140 for further details). The 7842 is capable of operating in multidrop applications. Each RS-422 driver/receiver pair is associated with a set of jumpers (W19, 20, 22, 23) that enable the drivers and receivers to be connected together. The RS-422 interface meets RS-485 standards, which allow up to 32 driver/receiver pairs to be placed on a single line. To configure channel A for multidrop, connect W19 pins 1-2 and 3-4. For channel B, connect W22 pins 1-2 and 3-4. For synchronous applications, connect channel A clocks (W20 pins 1-2 and 3-4) and channel B clocks (W23 pins 1-2 and 3-4)

Termination for the RS-422 lines is provided by 100 ohm resistors. If a different value is required to match line impedance, replace these resistors (R28-R31). In multidrop applications, termination is necessary only at the two extreme line locations. Other termination resistors should be removed since they will appear in parallel and reduce the signal.

Block Transfers

Channel A or channel B of the SIO can be dedicated to block I/O transfers. The channel A wait/ready (W/RDY*) output is connected to the Z80A wait (WAIT*) input by installing jumper W8 pins 1-2 and W/RDY B* is connected to the Z80A wait input by installing jumper W8 pins 3-4. Channels A and B cannot be selected together. Either W8 pins 1-2 or W8 pins 3-4 can be installed, but not both simultaneously.

PRIORITY CHAIN

The 7842 makes use of the STD BUS interrupt priority chain. Priority is based on position in the card rack with the highest priority on the right. (Note that some manufacturers make card racks with highest priority on the left.) The 7842 and other cards in a system which make use of the priority chain should be installed in the card rack according to their priority.

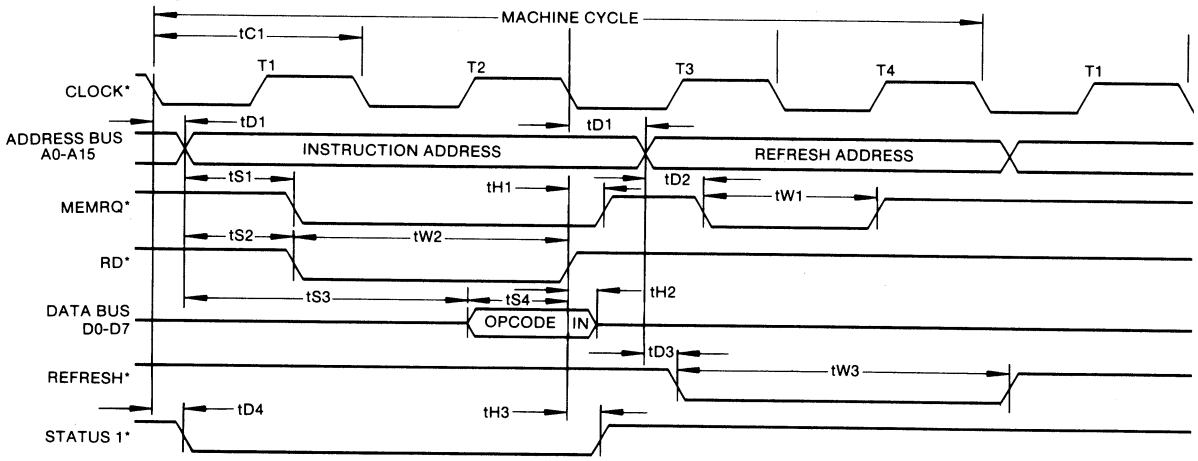
The 7842 has a fixed priority for the various interrupt sources on the card but, as a card, it may be higher or lower than other cards.

Cards which use the priority chain must be placed next to each other with no unused card slots in between. Other cards which do not use the priority chain may be installed between cards using the priority chain, but these other cards must pass on the priority chain: They must connect pins 51 and 52 (priority chain out and priority chain in) together onboard. This is standard practice on Pro-Log STD BUS cards.

The cards which use the priority chain, as a group, can be placed anywhere in the card rack and other cards can be placed anywhere on either side of the group. Other cards may have empty slots in between them or may be placed in the group provided they pass on the priority chain.

SPECIFICATIONS

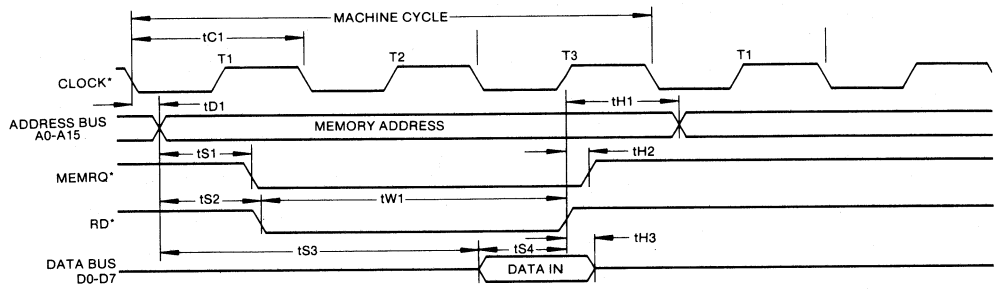
Figures 2-17 through 2-33 on the following pages give specifications for the 7842 card.



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	Address delay from CLOCK*		130		165
tD2	MEMRQ* delay from Refresh Address valid		115		155
tD3	REFRESH* delay from Refresh Address valid		40		55
tD4	STATUS 1* delay from CLOCK*		120		150
tS1	Address setup before MEMRQ* active	30		90	
tS2	Address setup before RD* active	80		150	
tS3	Address setup before DATA valid	265		530	
tS4	Data setup before RD* inactive	125		155	
tH1	MEMRQ* hold time after RD* inactive	0		0	
tH2	Data hold time after RD* inactive	0		0	
tH3	STATUS 1* hold time after RD* inactive	0		15	
tW1	MEMRQ* pulse width (REFRESH)	195		335	
tW2	RD* pulse width	335		530	
tW3	REFRESH* pulse width	460		735	

All times given in nanoseconds

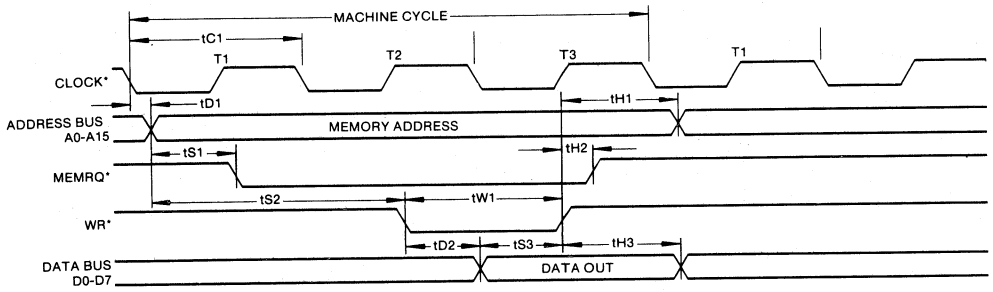
Figure 2-17. Opcode Fetch and Refresh Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	Address delay from CLOCK*		130		165
tS1	Address setup before MEMRQ* active	30		90	
tS2	Address setup before RD* active	80		150	
tS3	Address setup before data valid	405		730	
tS4	Data setup before RD* inactive	140		165	
tH1	Address hold time after RD* inactive	45		125	
tH2	MEMRQ* hold time after RD* inactive	0		0	
tH3	Data hold time after RD* inactive	0		0	
tW1	RD* pulse width	465		760	

All times given in nanoseconds

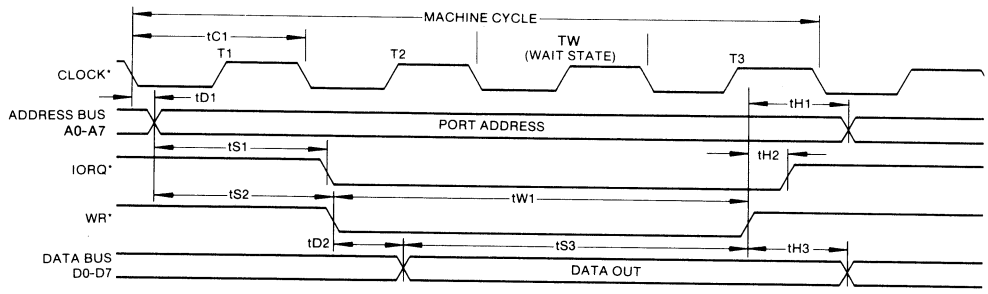
Figure 2-18. Memory Read Timing Other Than Opcode Fetch (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	Address delay from CLOCK*		130		165
tD2	Data delay from WR* active		75		75
tS1	Address setup before MEMRQ* active	30		90	
tS2	Address setup before WR* active	215		435	
tS3	Data setup before WR* inactive	150		290	
tH1	Address hold time after WR* inactive	45		125	
tH2	MEMRQ* hold time after WR* inactive	0		0	
tH3	Data hold time after WR* inactive	0		0	
tW1	WR* pulse width	195		335	

All times given in nanoseconds

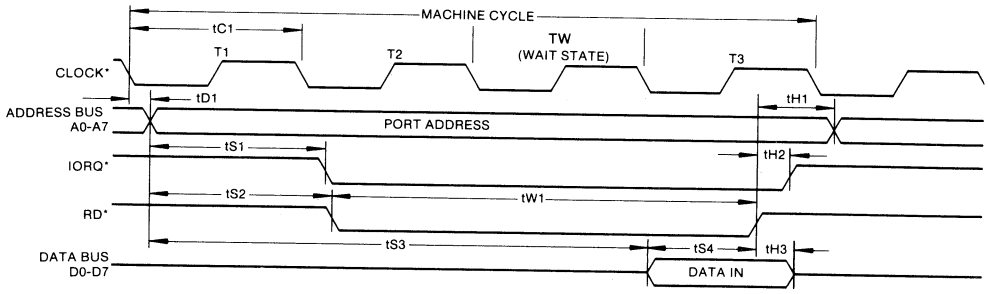
Figure 2-19. Memory Write Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	Address delay from CLOCK*		130		165
tD2	Data delay from WR* active		75		75
tS1	Address setup before IORQ* active	160		300	
tS2	Address setup before WR* active	185		310	
tS3	Data setup before WR* inactive	555		930	
tH1	Address hold time after WR* inactive	45		125	
tH2	IORQ* hold time after WR* inactive	0		0	
tH3	Data hold time after WR* inactive	0		0	
tW1	WR* pulse width	605		985	

All times given in nanoseconds

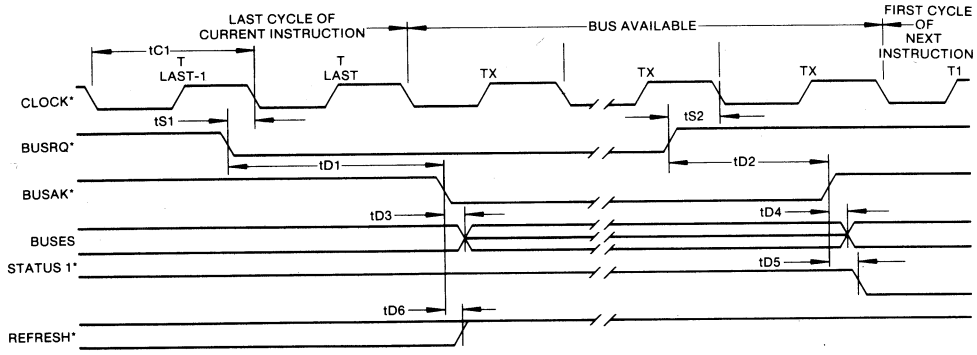
Figure 2-20. Output Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	Address delay from CLOCK*		130		165
tS1	Address set up before IORQ* active	160		300	
tS2	Address set up before RD* active	200		330	
tS3	Address set up before DATA valid	650		1120	
tS4	Data set up before RD* inactive	145		170	
tH1	Address hold time after RD* inactive	45		125	
tH2	IORQ* hold time after RD* inactive	0		0	
tH3	Data hold time after RD* inactive	0		0	
tW1	RD* pulse width	590		970	

All times given in nanoseconds

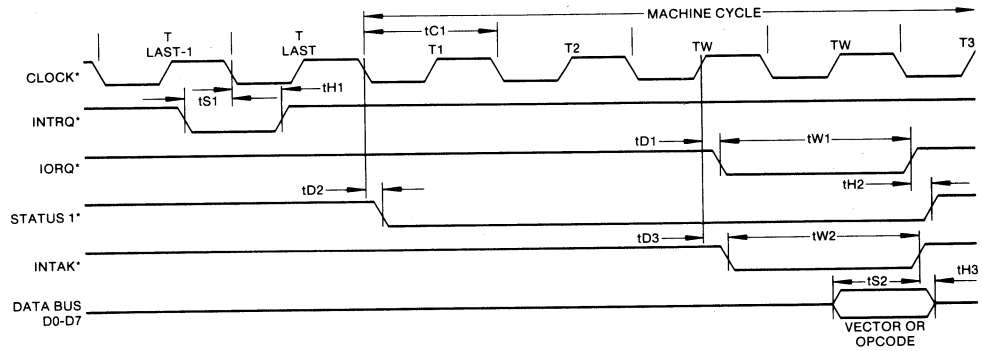
Figure 2-21. Input Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	BUSAK* delay from BUSRQ* active		455		655
tD2	BUSAK* inactive delay from BUSRQ* inactive		345		455
tD3	Buses delay after BUSAK* active		50		50
tD4	Buses delay after BUSAK* inactive		190		270
tD5	STATUS 1* delay after BUSAK* inactive		205		295
tD6	REFRESH* delay after BUSAK* active		65		75
tS1	BUSRQ* active set up before last time state	70		100	
tS2	BUSRQ* inactive set up before last time state	70		100	

All times given in nanoseconds

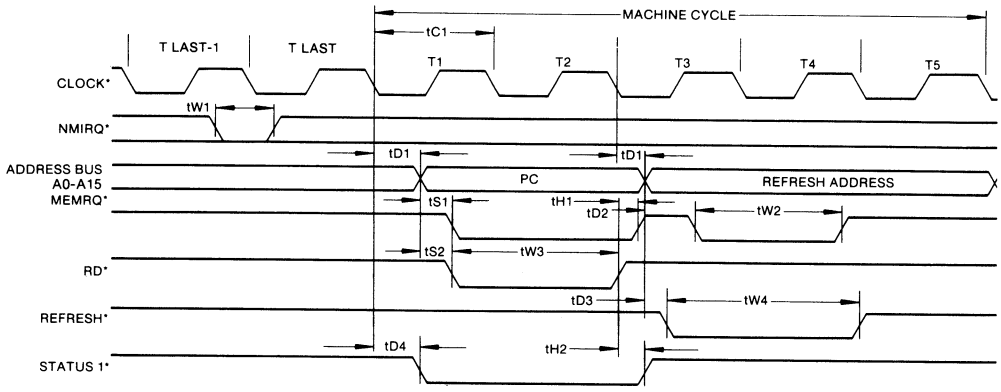
Figure 2-22. Bus Request Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	IORQ* delay from CLOCK*		120		130
tD2	STATUS 1* delay from CLOCK*		105		150
tD3	INTAK* delay from CLOCK*		140		165
tS1	INTRQ* set up before last time state	120		120	
tS2	Data set up before INTAK* inactive	85		115	
tH1	INTRQ* hold time after last time state	0		0	
tH2	STATUS 1* hold time after IORQ*	0		10	
tH3	Data hold time after INTAK* inactive	0		0	
tW1	IORQ* pulse width	340		550	
tW2	INTAK* pulse width	320		530	

All times given in nanoseconds

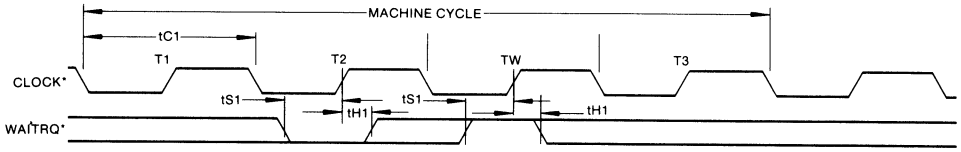
Figure 2-23. Interrupt Request Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tD1	Address delay from CLOCK*		130		165
tD2	MEMRQ* delay from Refresh Address valid		115		155
tD3	REFRESH* delay from Refresh Address valid		40		55
tD4	STATUS 1* delay from CLOCK*		120		150
tS1	Address set up before MEMRQ* active	30		90	
tS2	Address set up before RD* active	80		150	
tH1	MEMRQ* hold time after RD* inactive	0		0	
tH2	STATUS 1* hold time after RD* inactive	0		15	
tW1	NMIRQ* pulse width	60		60	
tW2	MEMRQ* pulse width (REFRESH)	195		335	
tW3	RD* pulse width	310		520	
tW4	REFRESH* pulse width	460		735	

All times given in nanoseconds

Figure 2-24. Nonmaskable Interrupt Request Timing (7842)



SYMBOL	PARAMETER	4 MHz		2.5 MHz	
		MIN	MAX	MIN	MAX
tC1	Clock cycle time	250		400	
tS1	WAITRQ* set up before CLOCK*	110		110	
tH1	WAITRQ* hold time after CLOCK*	0		0	

All times given in nanoseconds

Figure 2-25. Wait State Timing (7842)

Electrical Specifications

	RECOMMENDED OPERATING LIMITS			ABSOLUTE NONOPERATING LIMITS		
PARAMETER	MIN	TYP	MAX	MIN	MAX	UNITS
Supply Voltage	4.75	5.00	5.25	0	7.00	V
Supply Current(1)	N/A	1285	1715	N/A	N/A	mA
Aux +V supply voltage(2)	11.40	12.00	12.60	0	15.00	V
Aux +V supply current	N/A	19	30	N/A	N/A	mA
Aux -V supply voltage	-11.40	-12.00	-12.60	0	-15.00	V
Aux -V supply current	N/A	22	33	N/A	N/A	mA

NOTES:
(1) Without onboard memory.
(2) The auxiliary voltage source is necessary only if the channel A RS-232-C interface is used.

Environmental Specifications

	RECOMMENDED OPERATING LIMITS			ABSOLUTE NONOPERATING LIMITS		
PARAMETER	MIN	TYP	MAX	MIN	MAX	UNITS
Free Air Temperature	0	+25	+65	-40	+75	°C
Noncondensing relative humidity	5	N/A	95	5	95	%

Figure 2-26. Electrical and Environmental Specifications (7842)

Mechanical Specifications

The 7842 meets all general mechanical specifications of the STD BUS.

PARAMETER	MIN	TYP	MAX	UNITS
High-level output voltage		3.7		Volts
Low-level output voltage		1.1		Volts
High-level short-circuit current			180	mA
Low-level short-circuit current			-180	mA
Differential-input high-threshold voltage			0.2	Volts
Differential-input low-threshold voltage			-0.2	Volts

Figure 2-27. RS-422 Switching Characteristics

PARAMETER	MIN	TYP	MAX	UNITS
High-level output voltage(1)	8.2	9.5		Volts
Low-level output voltage(1)	-8.1	-9.4		Volts
High-level short-circuit current	6.0	10.0	12.0	mA
Low-level short-circuit current	-6.0	-10.0	-12.0	mA
High-level input threshold voltage			3.0	Volts
Low-level input threshold voltage			-3.0	Volts

(1)At termination resistance of 3.0K minimum

Figure 2-28. Channel A RS-232-C Switching Characteristics

PIN NUMBER				PIN NUMBER			
OUTPUT (LSTTL DRIVE)					OUTPUT (LSTTL DRIVE)		
INPUT (LSTTL LOADS)						INPUT (LSTTL LOADS)	
MNEMONIC							MNEMONIC
SEGMENT	7		2	1	OUT		GND

Figure 2-29. Segment Connector J1 Pin List (7842)

PIN NUMBER				PIN NUMBER			
OUTPUT (LSTTL DRIVE)					OUTPUT (LSTTL DRIVE)		
INPUT (LSTTL LOADS)						INPUT (LSTTL LOADS)	
MNEMONIC							MNEMONIC
CLOCK/INT 3	5		2	1	OUT		GND
CLOCK/INT 2	5		4	3	OUT		GND
COUNT/TIME 2		10	6	5	OUT		GND
CLOCK/INT 1	5		8	7	OUT		GND
COUNT/TIME 1		10	10	9	OUT		GND
W/RDY A*	1	9	12	11	OUT		GND
W/RDY B*	1	9	14	13	OUT		GND

*Low-level active logic

Figure 2-30. Counter/Timer Connector J2 Pin List (7842)

CONNECTOR J1 RS-232C

PIN NUMBER	SIGNAL FLOW	SIGNAL	PIN NUMBER	SIGNAL FLOW	SIGNAL
		2	1		
RCLK (DB)	OUT	4	3	IN/OUT	TXD* (BA)
		6	5	IN/OUT	RXD (BB)
RCLK(DD)	IN/OUT	8	7	IN/OUT	RTS (CA)
		10	9	IN/OUT	CTS (CB)
		12	11	IN/OUT	DSR (CC)
DTR (CD)	IN/OUT	14	13	OUT	SIGNAL GND (AB)
		16	15		
		18	17		
		20	19		
TCLK(DA)	IN/OUT	22	21		
		24	23		
		26	25		

Figure 2-31. RS-232-C Serial Connector J3 Pin List (7842)

PIN NUMBER	SIGNAL FLOW	SIGNAL	PIN NUMBER	SIGNAL FLOW	SIGNAL
TXC* -	OUT	2	1	OUT	TXC* +
TXD -	OUT	4	3	OUT	TXD +
GND	OUT	6	5	OUT	GND
RXD +	IN	8	7	IN	RXD -
RXC* +	IN	10	9	IN	RXC* -

Figure 2-32. RS-422 Serial Connector J4, J5 Pin List (7842)

PIN NUMBER	OUTPUT (LSTTL DRIVE)	INPUT (LSTTL LOADS)	MNEMONIC	PIN NUMBER	OUTPUT (LSTTL DRIVE)	INPUT (LSTTL LOADS)	MNEMONIC
2	IN		+5V	1		IN	+5V
4	IN		GROUND	3		IN	GROUND
6			-5V	5			-5V
8	1	55	D7	7	55	1	D3
10	1	55	D6	9	55	1	D2
12	1	55	D5	11	55	1	D1
14	1	55	D4	13	55	1	D0
16	1	55	A15	15	55	1	A7
18	1	55	A14	17	55	1	A6
20	1	55	A13	19	55	1	A5
22	1	55	A12	21	55	1	A4
24	1	55	A11	23	55	1	A3
26	1	55	A10	25	55	1	A2
28	1	55	A9	27	55	1	A1
30	1	55	A8	29	55	1	A0
32	1	55	RD*	31	55	1	WR*
34	1	55	MEMRQ*	33	55		IORQ*
36		50	MEMEX	35	OUT		IOEXP (GROUND)
38		55	MCSYNC*	37	55		REFRESH*
40			STATUS 0*	39	55		STATUS 1*
42	5		BUSRQ*	41	55		BUSAK*
44	5		INTRQ*	43	55		INTAK*
46	5		NMIRQ*	45		5	WAITRQ*
48	(1)		PBRESET*	47	55		SYSRESET*
50			CNTRL*	49	55		CLOCK*
52	5		PCI	51	55		PCO
54	IN		AUX GND	53		IN	AUX GND
56	IN		AUX -V	55		IN	AUX +V

*Low-level active logic
(1) Switch input

Figure 2-33. STD BUS Connector Pin List (7842)

Reset

Power-on reset on the 7842 uses a voltage comparator to ensure that VCC rises to an acceptable operating level before the onboard reset line and the STD BUS SYSRESET* line are released. Once VCC has reached an acceptable level, a delay of 40 milliseconds maximum occurs before reset is released. This delay gives the onboard clock time to begin functioning.

Pushbutton reset is activated by the STD BUS line PBRESET*. The line is debounced approximately 5 milliseconds when it goes low and approximately 20 milliseconds when it goes high. The resulting reset signal is synchronized with the Z80A M1* signal and is 30 nanoseconds typical to 50 nanoseconds maximum in duration. This avoids disturbing the refresh cycle of dynamic memories. If the Z80A does not generate an M1* signal, a reset pulse occurs after a delay of approximately 400 nanoseconds.

INTRODUCTION

This section covers the operation of the 7842 serial ports, counter/timer, and memory expansion line. All of these functions are controlled through I/O ports.

MEMORY EXPANSION

Two lines on the 7842 can be used for memory expansion, the STD BUS MEMEX line and the external signal Segment. How these lines affect memory on the 7842 is explained under the Memory heading of section 2 of this manual, page 2-6.

The Segment line is an input, and cannot be controlled by the 7842. The MEMEX line is controlled by bit 0 of the MEMEX control port (figure 2-12, page 2-14). Port configuration as inverting or noninverting is controlled by jumper W11 (figure 2-5, page 2-4).

SERIAL COMMUNICATION CHANNELS

SIO Internal Structure

Operation of the serial ports depends upon the contents of the SIO internal control (write) and status (read) registers. Figure 3-1 shows the structure of the SIO channel B registers. Write register 2 (WR2) and read register 2 (RR2) are not included in channel A. The remaining registers of the two channels are identical. Figures 3-2 and 3-3, pages 3-3 and 3-4, show the basic functions of the registers.

In addition to the control and status registers, each channel has a transmit data (XMT) register and a receive data (RCV) register. Each channel occupies two I/O addresses with WR0 and RR0 at one address and XMT and RCV data at the other address. The other seven control registers and two status registers are addressed through WR0.

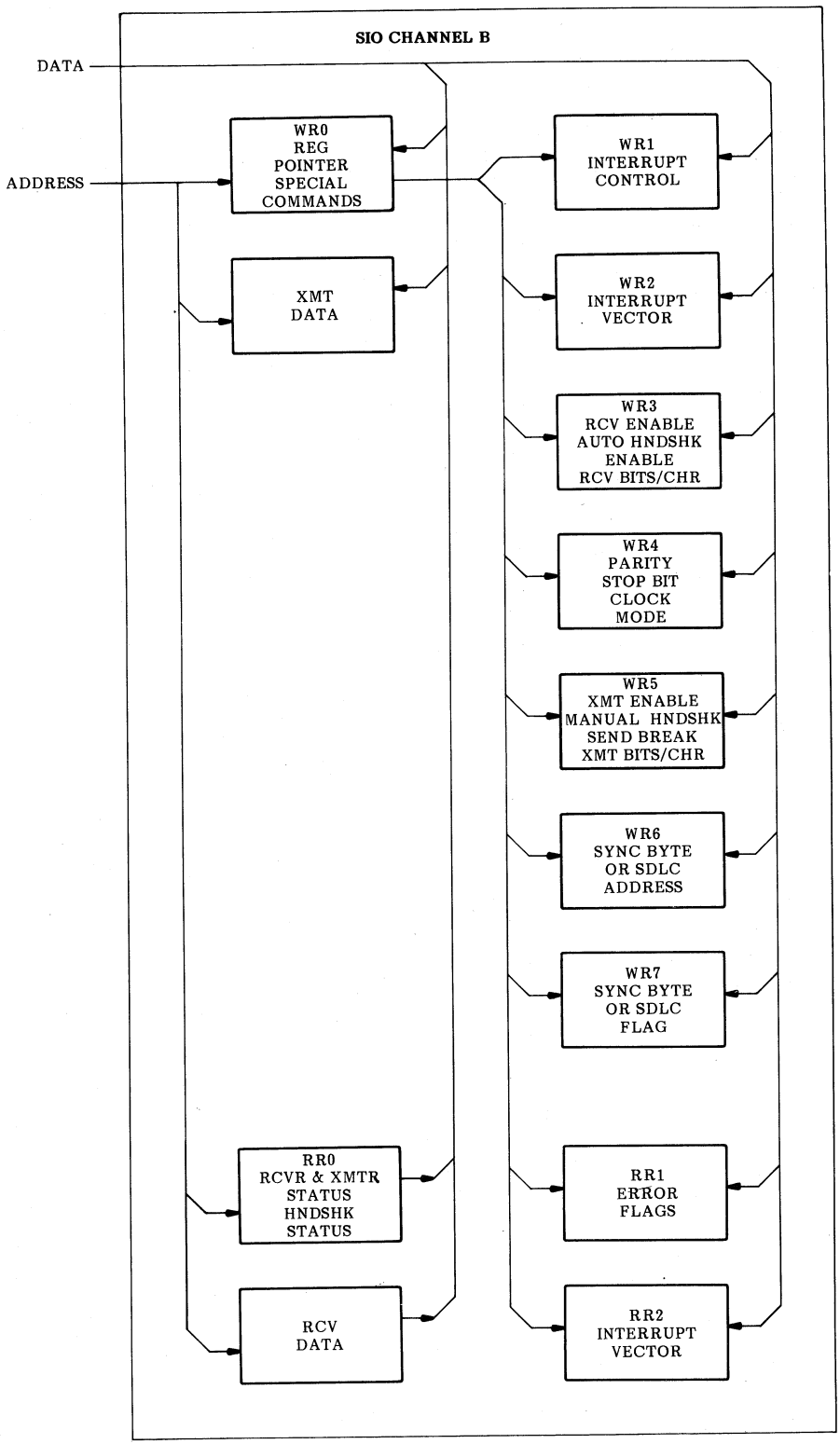
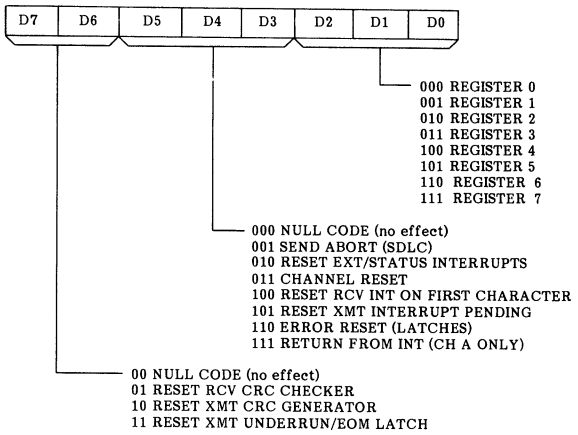
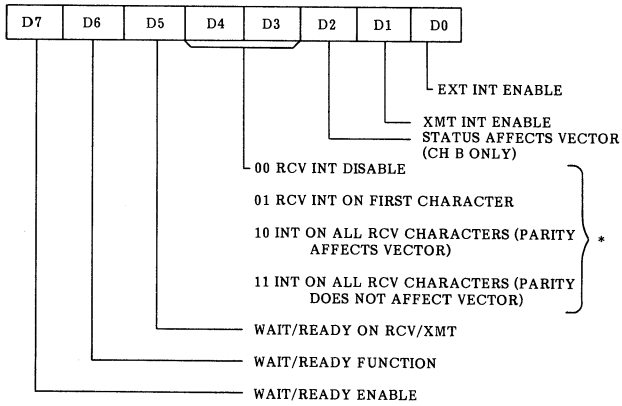


Figure 3-1. SIO Register Structure (7842)

WRITE REGISTER 0

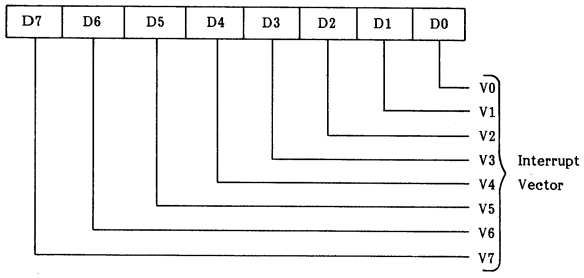

D7 D6 D5 D4 D3 D2 D1 D0
000 REGISTER 0
001 REGISTER 1
010 REGISTER 2
011 REGISTER 3
100 REGISTER 4
101 REGISTER 5
110 REGISTER 6
111 REGISTER 7
000 NULL CODE (no effect)
001 SEND ABORT (SDLC)
010 RESET EXT/STATUS INTERRUPTS
011 CHANNEL RESET
100 RESET RCV INT ON FIRST CHARACTER
101 RESET XMT INTERRUPT PENDING
110 ERROR RESET (LATCHES)
111 RETURN FROM INT (CH A ONLY)
00 NULL CODE (no effect)
01 RESET RCV CRC CHECKER
10 RESET XMT CRC GENERATOR
11 RESET XMT UNDERRUN/EOM LATCH

WRITE REGISTER 1

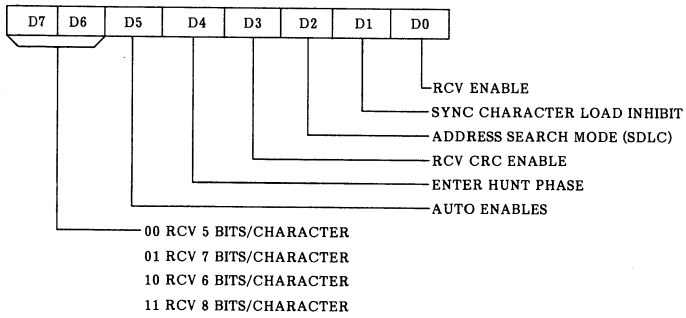

D7 D6 D5 D4 D3 D2 D1 D0
EXT INT ENABLE
XMT INT ENABLE
STATUS AFFECTS VECTOR (CH B ONLY)
00 RCV INT DISABLE
01 RCV INT ON FIRST CHARACTER
10 INT ON ALL RCV CHARACTERS (PARITY AFFECTS VECTOR)
11 INT ON ALL RCV CHARACTERS (PARITY DOES NOT AFFECT VECTOR)
*
WAIT/READY ON RCV/XMT
WAIT/READY FUNCTION
WAIT/READY ENABLE

*OR ON SPECIAL RCV CONDITION

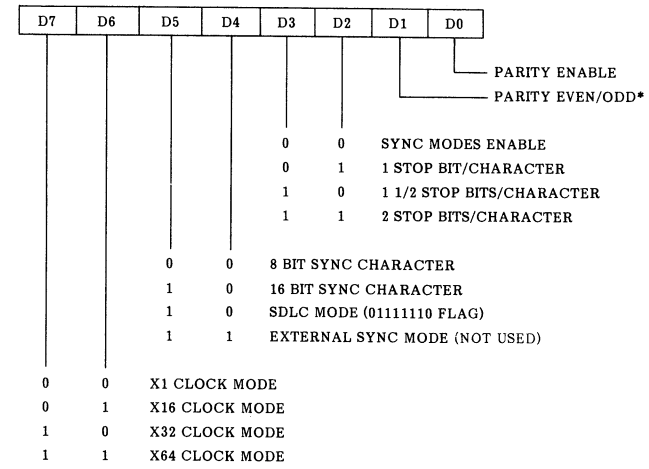
WRITE REGISTER 2 (Channel B only)



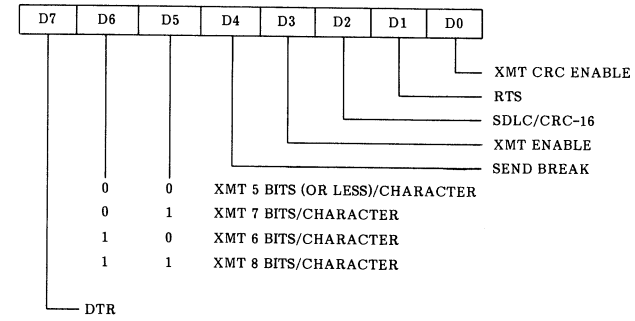
WRITE REGISTER 3



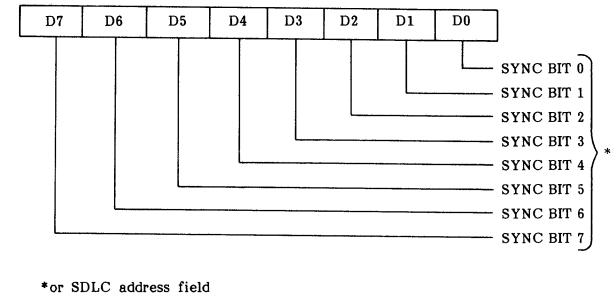
WRITE REGISTER 4


D7 D6 D5 D4 D3 D2 D1 D0
PARITY ENABLE
PARITY EVEN/ODD*
0 0 SYNC MODES ENABLE
0 1 1 STOP BIT/CHARACTER
1 0 1 1/2 STOP BITS/CHARACTER
1 1 2 STOP BITS/CHARACTER
0 0 8 BIT SYNC CHARACTER
1 0 16 BIT SYNC CHARACTER
1 0 SDLC MODE (01111110 FLAG)
1 1 EXTERNAL SYNC MODE (NOT USED)
0 0 X1 CLOCK MODE
0 1 X16 CLOCK MODE
1 0 X32 CLOCK MODE
1 1 X64 CLOCK MODE

WRITE REGISTER 5

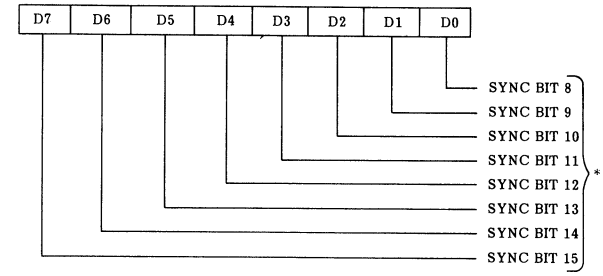


WRITE REGISTER 6



*or SDLC address field

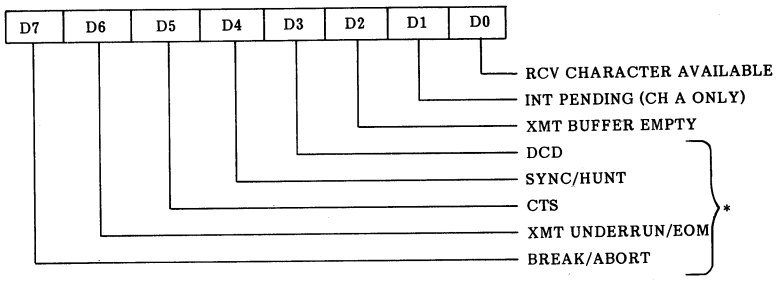
WRITE REGISTER 7



*or SDLC flag

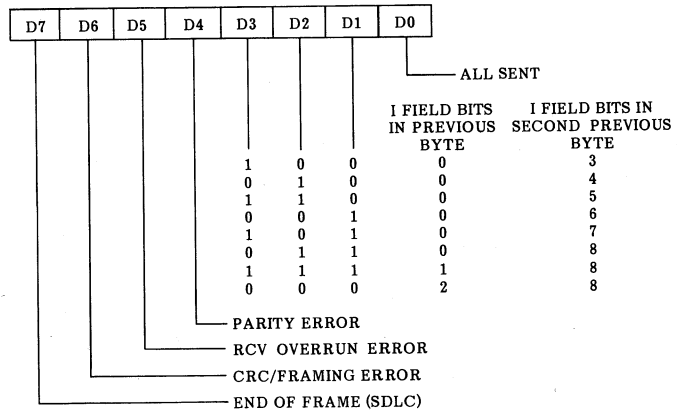
Figure 3-2. Write Register Bit Maps (7842)

READ REGISTER 0

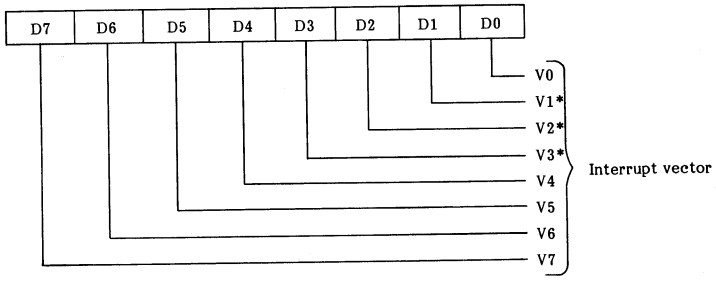


*Used with EXT/STATUS interrupt mode

READ REGISTER 1



READ REGISTER 2 (Channel B only)



* Variable if "Status Affects Vector" is programmed

Figure 3-3. Read Register Bit Maps (7842)

SIO Receive and Transmit Registers

The RCV data register is directly addressable and holds the data that is received over the serial interface. It holds up to three bytes in addition to the byte currently being received. The error flags in the read register are also four levels deep so the appropriate status information is available with each byte. RR1 should be read before the RCV data register as the stacks for both data and status are incremented when the data is read. If enabled, an interrupt occurs each time a character reaches the top of the stack. If there are no characters pending in the stack, a received character goes directly to the top of the stack.

If four complete bytes have been received since the last time the processor read a byte out of the RCV data register, the third byte will be overwritten. The fourth character, in the shift register, is loaded into the bottom of the stack of received characters as soon as it is completely received. If the Status Affects Vector mode is selected (discussed in the SIO Operation section, page 3-25), an interrupt is generated with a special vector indicating an error. This interrupt occurs when the character which overwrote the previous one reaches the top of the stack. An error flag is set in the status information associated with the fourth byte, indicating that it overwrote the previous byte.

In the Interrupt on First Received Character mode (interrupts can occur on the first character or every character), if an error occurs the data and status stacks are frozen, even if read, and will prevent access to further data until an error reset is performed. Error reset is explained in the WR0 description on page 3-7.

The XMT data register is directly addressable and holds the data to be transmitted. In addition to the XMT data register, there is a 20-bit shift register for the byte currently being transmitted. This allows the processor to give the SIO one byte while it is transmitting another, thereby eliminating delays between characters.

An interrupt and/or flag indicates that the XMT data register is empty and is able to accept another character. The All Sent flag in RR1 indicates that actual transmission has been completed. The All Sent condition does not generate an interrupt.

SIO Control Registers

The SIO contains eight control registers for channel B and seven for channel A. WR2 which holds the interrupt vector for both channels is only in the channel B register set. All control registers require two bytes to be programmed except for WR0. The first byte contains the register select code in the least significant 3 bits and, if the code is for WR0, the remainder of the byte is the control word. For the remaining registers, the control byte is the next word. The first byte written into each channel after a reset (either with a channel reset command or hardware system reset) is directed to WR0. Figure 3-4 lists the functions assigned to each control register. Figures 3-5 through 3-12, pages 3-7 through 3-18, give detailed descriptions of these registers. All references to pin states are as seen at the SIO. The state programmed into a bit is the state that appears at the pin. The RS-232-C interface inverts the signal. In most systems the equipment at the other end of the serial port will also have an RS-232-C inversion.

WR0	Register pointers, CRC initialize, initialization commands for the various modes, etc.
WR1	XMT/RCV interrupt and data transfer mode definition
WR2	Interrupt vector (channel B only)
WR3	Receive parameters and control
WR4	XMT/RCV miscellaneous parameters and modes
WR5	Transmit parameters and controls
WR6	Sync character or SDLC address field
WR7	Sync character or SDLC flag

Figure 3-4. SIO Control Register Functions

Write Register 0

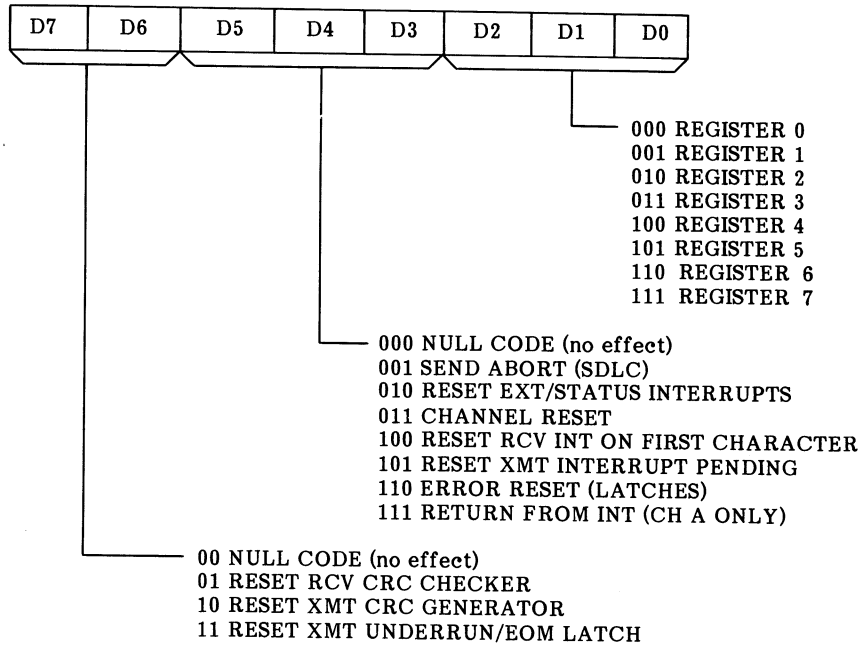


Figure 3-5. Write Register 0 Bit Map (7842)

D7-D6 **Reset Bits**

These bits specify three reset modes: reset receive CRC check, reset transmit CRC generator, and reset sending transmit underrun/end of message latch.

D5-D3 **Command Bits**

COMMAND 0 This is a "no operation" command which can be issued while setting the register pointer when no other effect is desired.

COMMAND 1 A sequence of 8 to 13 ones is generated. It is used only with the SDLC mode.

COMMAND 2 After the status bits of RR0 have been latched by a change of one of the external/status bits, this command re-enables the latch of RR0. If an external/status interrupt occurs, command 2 is issued to re-enable RR0, which allows external/status interrupts to occur. The latching allows capture of a short pulse on the input pins and holds the pulse until the CPU can read it.

COMMAND 3 This command performs the same function as a system reset, but on a single channel only. The channel A reset also resets interrupt prioritization logic. All control registers must be rewritten after a channel A reset, then four system (I) clock cycles should be allowed for reset before the next SIO access.

COMMAND 4 If the Interrupt Only on First Received Character mode is selected (D4-D3, WR1), it must be reactivated after each complete message is received. This mode is used during block transfers to prevent interrupts during the block transfer: the only interrupt is at the start of the block.

COMMAND 5 The transmitter interrupts when it becomes empty if transmit interrupt was enabled at D0 of WR1. This command resets the transmitter interrupt pending until after the next character has been loaded into the transmitter. This is used during periods when no characters are to be sent.

COMMAND 6 Parity and overrun errors are latched in read register 1 until reset by this command. The latching allows errors occurring during block transfers to be examined at the end of the block (block transfers are discussed in the SIO Operation portion of this section, page 3-25).

COMMAND 7 This command has the same effect as a return from interrupt command performed by the processor (return from interrupt is explained in the SIO Operation portion of this section, page 3-25). This command can only be issued to channel A, but affects the whole SIO by resetting the interrupt under service latch of the SIO device under service, allowing lower priority devices to interrupt.

D2-D0 Register Pointers

These bits indicate to which register the next write command is written. The first byte written into each channel after a reset (either with a channel reset command or hardware system reset) is directed to WR0 and the byte following a read or write to any register.

Write Register 1

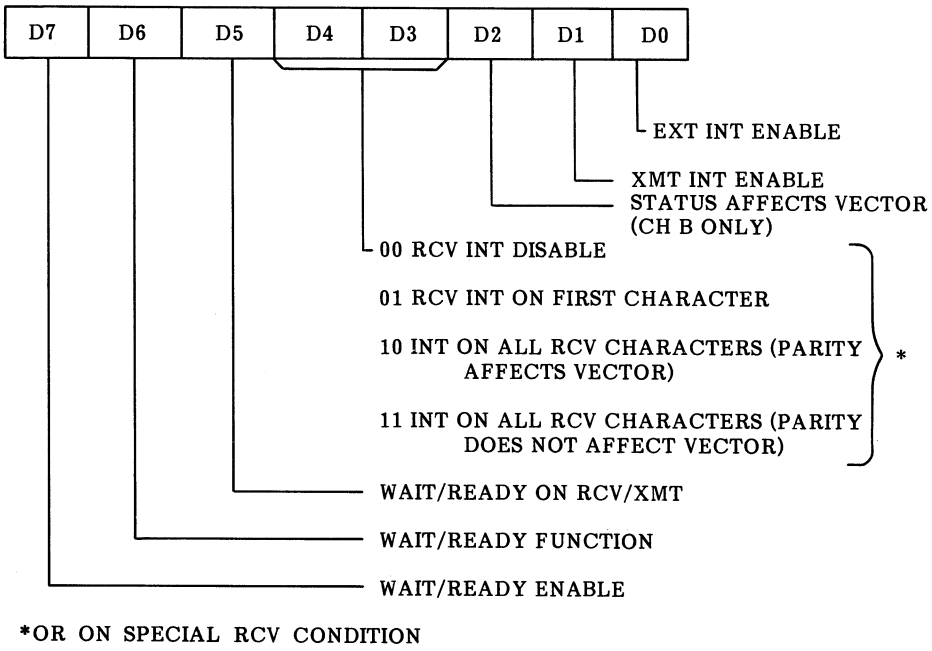


Figure 3-6. Write Register 1 Bit Map (7842)

D7 **Wait/Ready Enable**

The wait/ready pin for a channel remains high if programmed to ready mode by bit 6, or floating if programmed to wait mode by bit 6, until this enable bit is set to 1. Care must be taken when using wait in a dynamic RAM system; the wait period must not interfere with memory refresh.

D6 **Wait/Ready Function**

This bit selects whether the wait/ready A/B line acts as wait (0) or ready (1). Default condition is wait. As a wait line, it is generally used with Z80A block transfer instructions to synchronize the CPU to the SIO data rate. The wait line becomes active whenever the processor attempts to read or write data before the SIO has data in its receiver or can accept data in its transmitter. Care must be taken not to disturb dyanmic RAM refresh when using the wait line.

As a ready line, it is usually used with a DMA controller. It is active whenever the receiver has a character or whenever the transmitter is empty. The ready line is actively driven and occurs just after the positive edge of the system clock. The reset state for wait/ready pins is open active.

D5 **Wait/Ready on Receive/Transmit**

When the wait/ready line is enabled, this bit determines whether it is active when the receiver is empty (bit = 1) or when the transmit buffer is full (bit = 0).

D4-D3 **Receiver Interrupt Mode**

These bits specify the character available interrupt conditions. If the Status Affects Vector is set (D2), the vector generated depends on whether or not an error occurs. The special receive condition interrupt occurs if there is a receiver error, otherwise the receive character available interrupt occurs.

D2 (Channel B Only) **Status Affects Vector**

This bit can only be programmed in the channel B register. When this mode is selected, the interrupt vector written by the SIO during an interrupt acknowledge cycle will be variable according to the following table:

	V_3	V_2	V_1	
Ch B	0	0	0	Ch B Transmit Buffer Empty
	0	0	1	Ch B External/Status Change
	0	1	0	Ch B Receive Character Available
	0	1	1	Ch B Special Receive Condition *
Ch A	1	0	0	Ch A Transmit Buffer Empty
	1	0	1	Ch A External/Status Change
	1	1	0	Ch A Receive Character Available
	1	1	1	Ch A Special Receive Condition

The variable bits are inserted into the interrupt vector which may be read at RR2. The remaining bits of the vector at RR22 come from the vector written to WR2. If the Status Affects Vector bit equals 0, the vector programmed at WR2 is returned unchanged at RR2. Although the Status Affects Vector can only be written to channel B, it is used by both channels.

D1 **Transmitter Interrupt Enable**

This bit allows interrupts whenever the transmitter buffer becomes empty. This interrupt generates the Transmit Buffer Empty vector of RR2 if the Status Affects Vector bit is programmed (D2, WR1).

D0 **External Interrupt Enable**

This bit allows interrupts to occur as a result of transitions on the DCD* or CTS* lines (Channel A RS-232C only), or as a result of a break condition (read at RR0), or the beginning of sending CRC or sync characters. These interrupts show in the external/status change vector of RR2 if the Status Affects Vector bit is programmed (D2, WR1).

Write Register 2 (Channel B only)

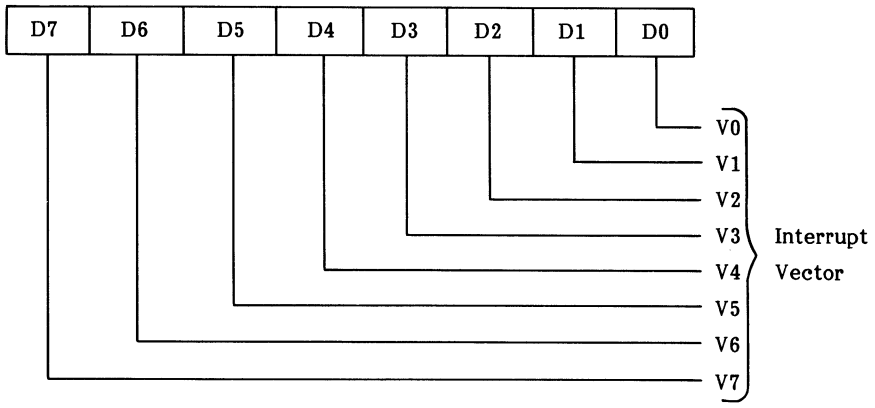


Figure 3-7. Write Register 2 Bit Map

The interrupt vector is written to this register and is read back at RR2. This register can only be written in channel B, but the vector is used by both channels. The vector at RR2 is unchanged if the status is used by both channels. The vector at RR2 is unchanged if the Status Affects Vector bit of WR1 is a zero. Otherwise, status information is programmed into the vector that is copied into RR2.

Write Register 3

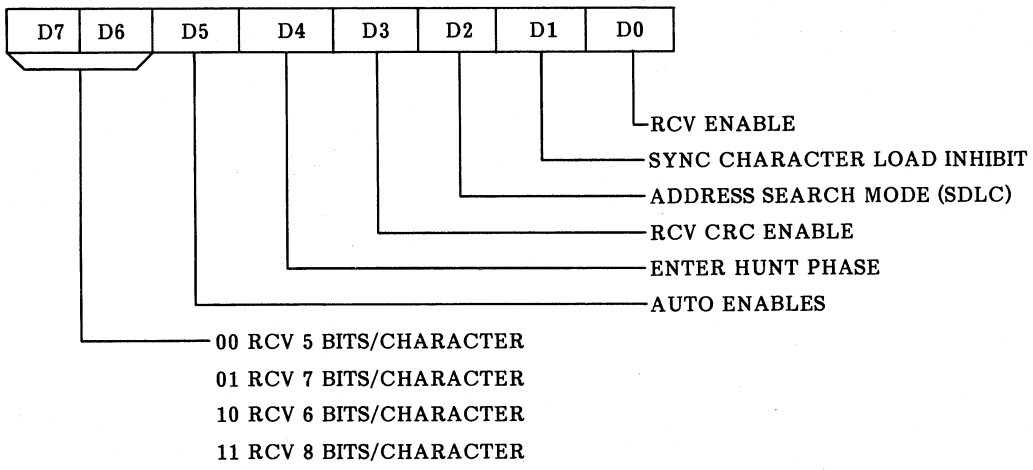


Figure 3-8. Write Register 3 Bit Map

D7-D6 **Receiver Bits/Character**

These bits determine the number of serial receive bits that are assembled to form a character. An additional parity bit is received if parity is enabled at D0 of WR4.

D5 **Auto Enables**

If this mode is selected, the DCD* and CTS* lines for the channel are receiver and transmitter enables, respectively. If the mode is not selected, the DCD* and CTS* are inputs to bits in RR0. Transitions on these lines can cause an interrupt if external interrupts are enabled at D0 of WR1. The Auto Enables mode can be used only with the channel A RS-232-C interface.

D4 **Enter Hunt Mode**

If character synchronization is lost or, if in the SDLC mode, the incoming message is not needed, the Hunt mode may be re-entered by setting this bit.

D3 **Receiver CRC Enable**

When this bit is set, CRC calculation begins at the start of the last character transferred from the receive register to the buffer stack. This occurs regardless of the number of characters in the stack.

D2 **Address Search Mode**

When this bit is set in the SDLC mode selected, no interrupts are generated unless an address match (programmed or global) occurs.

D1 **Sync Character Load Inhibit**

When this bit is set, sync characters preceding a message are not loaded into the receive buffers. The CRC calculation is not terminated if no sync characters are generated.

D0 **Receiver Enable**

This bit enables the receiver and is normally left active throughout operation. If the receiver is disabled while receiving a character, the character is lost and no interrupt occurs.

Write Register 4

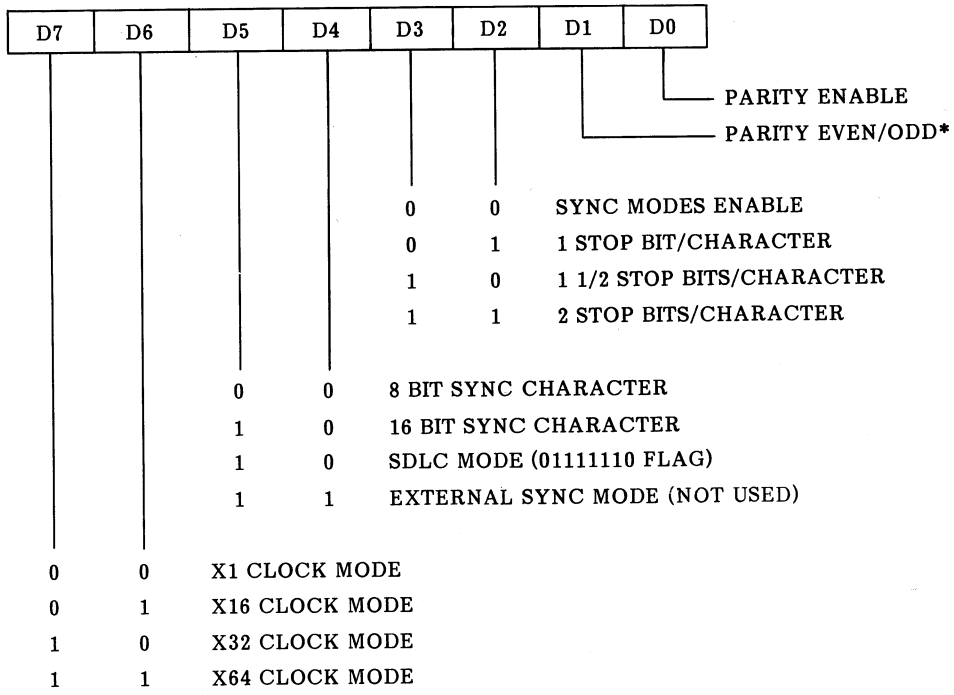


Figure 3-9. Write Register 4 Bit Map

D7-D6 **Clock Rate**

The clock rate specifies the multiplier between clock and data rates. The same multiplier is used for both the receiver and transmitter. In all modes, the system clock must be at least five times the data rate. If the X1 clock rate is selected, bit synchronization must be accomplished externally.

D5-D4 **Sync Modes**

These bits select character synchronization options. The external sync mode is not available on the 7842 card.

D3-D2 **Stop Bits**

These bits determine the number of stop bits added to each character sent. The receiver always checks for one stop bit.

D1 **Parity Even/Odd**

If parity is specified by bit D0, this bit determines whether the parity is sent or checked as even (bit = 1) or odd (bit = 0).

D0 **Parity**

When this bit is set, an additional bit position (in addition to those specified in the bits/character control of WR3 or 4) is added to transmitted data and is expected in receive data.

Write Register 5

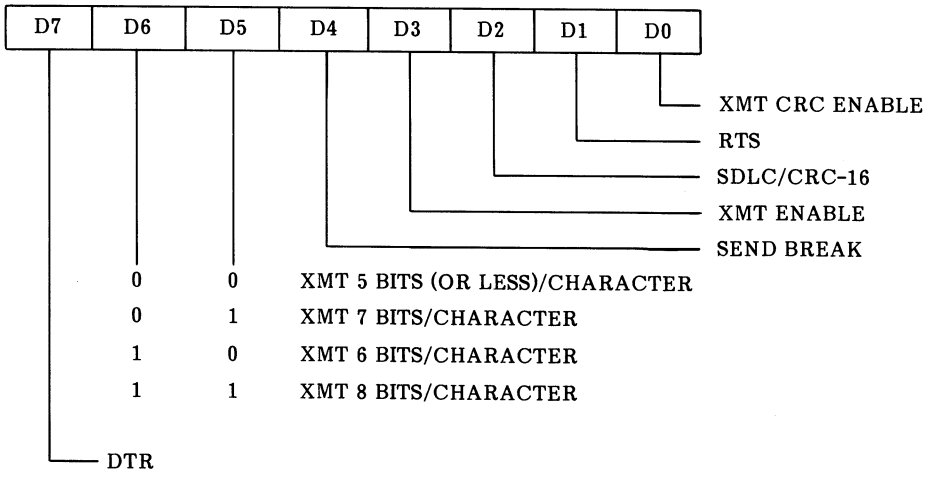


Figure 3-10. Write Register 5 Bit Map

D7 **Data Terminal Ready**

This is the control bit for the DTR* output pin for the channel. When set, DTR* is active low. When reset (0), DTR* is inactive high. This output is used to control the enables of the RS-422 drivers and is a general-purpose output for the RS-232-C interface.

D6-D5 **Transmit Bits/Character**

These bits determine the number of bits that are sent from each byte transferred to the transmit buffer. An additional parity bit is sent if parity is enabled at D0 of WR4. Bits are assumed to be right justified. If the 5 bits/character mode is selected, the unused bits of each data byte must be zero.

D4 **Send Break**

When set, this bit forces the TxD pin for the channel to spacing low, regardless of any data being transmitted. When reset, the TxD pin for the channel is released.

D3 **Transmit Enable**

This bit enables the transmitter. Data is not transmitted and the transmit data pin remains high until the transmitter is enabled. If the transmitter is disabled while sending, the character being sent will be completed. An interrupt still occurs when the XMT Data register becomes empty. Only the character being sent is completed. If one is pending in the XMT Data register, it remains pending. CRC characters are not sent if the transmitter is disabled during CRC transmission.

D2 **SDLC/CRC**

This bit selects the CRC code used by the transmitter and receiver. When the bit is set, the CRC-16 polynomial is used and the CRC registers are reset to 0. When D2 is reset, the SDLC polynomial $X^{16} + X^5 + 1$ is used.

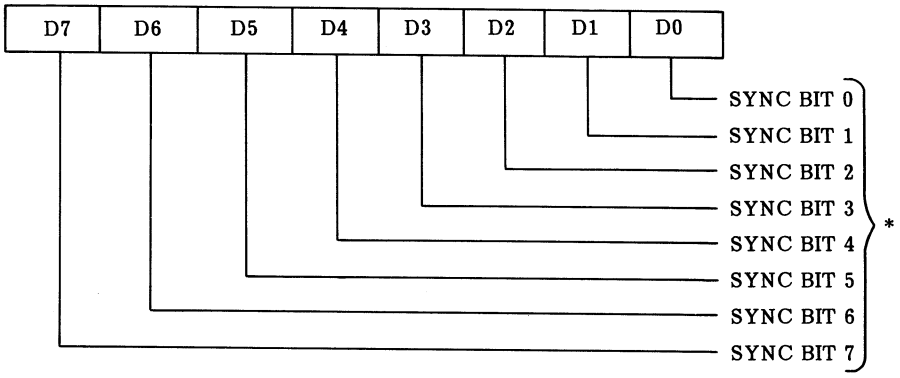
D1 **Request to Send**

This is the control bit for the RTS* output pin for the channel. When set, the RTS* line goes active low. When reset to 0, the RTS* pin is inactive high only after the transmitter is empty. This output is only used with the channel A RS-232-C interface.

D0 **Transmit CRC Enable**

If this bit is set at the time of loading the character from the transmit buffer to the transmit shift register, the CRC is calculated on the character. The CRC is not automatically sent when the transmitter is empty unless this bit is set.

Write Register 6



*or SDLC address field

Figure 3-11. Write Register 6 Bit Map.

WR6 contains the first 8 bits of a BISYNC sequence. It must contain the check address (if used) for SDLC mode, and the sync character in the 8-bit sync mode.

Write Register 7

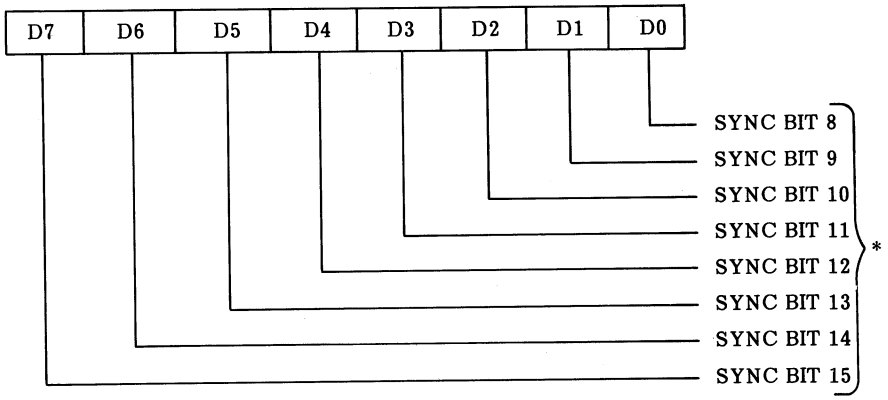


Figure 3-12. Write Register 7 Bit Map

This register contains an 8-bit sync character, or the second byte of a 16-bit synchronization sequence. In SDLC mode, it must contain the flag 01111110.

SIO Status Registers

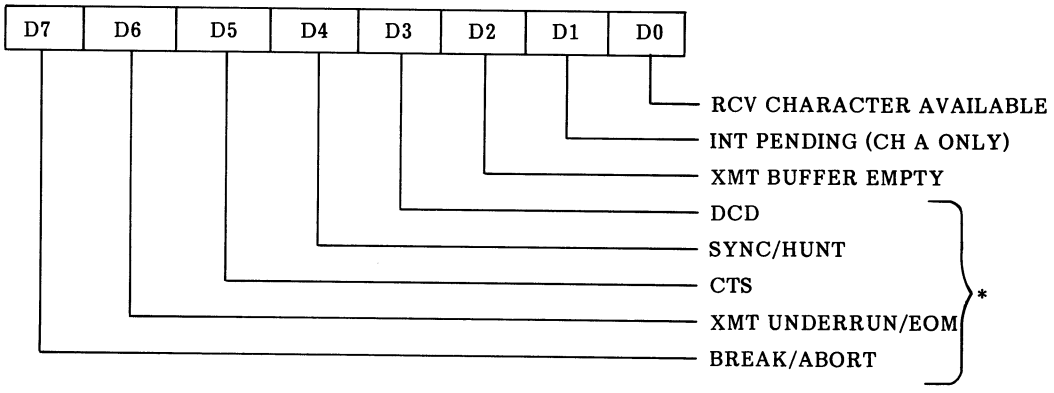
The SIO contains three read registers for channel B and two read registers for channel A that hold status information. These registers contain error conditions, an interrupt vector, and standard communications interface signals. RR2 contains the interrupt vector and is only included in the channel B register set. All read registers require two bytes to be read except for RR0. The first byte must be a write to WR0, with the register select code in the least significant three bits. The next byte is read from the read register. The byte following a read or write to any register is directed to register 0. Figure 3-13 lists the functions assigned to each status register. A detailed description of each status register follows each of figures 3-14 through 3-16.

All references to pin states are as seen at the SIO. The state that appears at the pin is the state that is seen in the status bit. The RS-232-C interface inverts the signal. In most RS-232-C systems, the equipment at the other end of the serial port has an RS-232-C inversion.

RR0	Transmit/receive status, interrupt status
RR1	Error status, residue codes
RR2	Interrupt vector

Figure 3-13. Status Register Functions

Read Register 0



*Used with EXT/STATUS interrupt mode

Figure 3-14. Read Register 0 Bit Map

D7 **Break/Abort**

In asynchronous modes, this bit is set when a "break" (continuous zeros) is detected. After the Reset external/status interrupts command (command 2, WR0) has been executed, the break bit is reset when the break stops. If external status interrupts have been enabled (D0, WR1), the break causes interrupt on each transition of the external signal. An external/status change vector is sent if the Status Affects Vector bit is programmed (D2, WR1). In SDLC mode, an abort sequence (7 or more ones) sets this bit. It is not used in other synchronous modes.

D6 **Sending CRC/Sync**

The CRC is automatically sent when the transmitter is empty for the first time in a message. Interrupts are generated (if enabled) when D6 is set, but are not sent when reset. If the Transmitter Buffer Empty bit is not set when D6 is set, the CRC character is being transmitted. If both Transmit Buffer Empty and Sending CRC/Sync are set, sync characters are being transmitted.

D5 **Clear to Send**

This bit shows the status of the channel CTS* pin at the time of the first change of any of the five external/status bits (DCD, CTS, Sync/Hunt, Break/Abort, or Sending CRC/Sync) after power on, channel reset, or an external/status interrupt (command 2). The change of one of these bits latches the status bits of RR0. To get the current state of the CTS* pin, this bit must be read immediately following a reset external/status interrupt command (command 2, WR 0). A transition at the CTS* pin causes an interrupt if external interrupts are enabled at D0 of WR1. An external/status change vector is sent if the Status Affects Vector bit is programmed (D2, WR1). The CTS* input is only used with the RS-232-C interface.

D4 **Sync/Hunt**

This bit is reset when character synchronization is achieved and is set by writing the Enter Hunt Mode bit.

D3 **Data Carrier Detect**

This bit shows the status of the DCD* pin at the time of the first change of any of the five external/status bits (DCD, CTS, Sync/Hunt, Break/Abort, or Sending CRC/Sync) after power on, channel reset, or an external/status interrupt (command 2). The change of one of these bits latches the status bits of RR0. To get the current state of the DCD* pin, this bit must be read immediately following a reset external/status interrupt command (command 2, WR0). A transition at the DCD* pin causes an interrupt if

external interrupts are enabled at D0 of WR1. An external/status change vector is sent if the Status Affects Vector bit is programmed (D2, WR1). The DCD* input is only used with the channel A RS-232-C interface.

D2 **Transmit Buffer Empty**

This bit is set whenever the channel transmit buffer is empty except during CRC transmission in synchronous mode. This causes an interrupt which shows in the transmit buffer empty vector of RR2 if the Status Affects Vector bit is programmed (D2, WR1).

D1 (Channel A Only) **Interrupt Pending**

Any interrupt condition present in the SIO sets this bit, but it can only be read from channel A. This bit is read as a 0 in channel B.

D0 **Receive Character Available**

This bit is set when a character is available in the receive buffer. The receiver ports are quadruply buffered (three storage registers and an input shift register) to allow additional time for the CPU to service an interrupt at the beginning of a block of high speed data. This bit is reset automatically when the receive buffers are all empty. Error status reflects the current word in the receive buffer. When the receive character bit is set, RR1 should be read and stored to avoid losing any error information. The data received can then be read. Error information does not need to be read with vectored interrupts, since a special vector is generated if there is an error. A receive character is available to be read a maximum of 13 system clock cycles from the RxC signal that samples the last bit of the data. Since the receiver port has three storage registers, three characters can be contained in the SIO without a read. A fourth character received without a read causes an overrun error in the last receiver storage register.

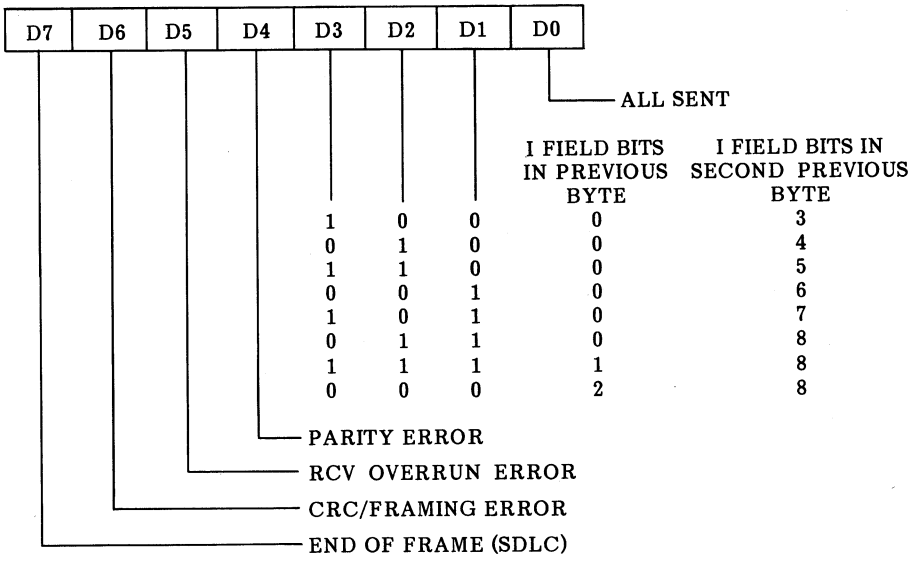


Figure 3-15. Read Register 1 Bit Map

D7 **End of Frame (SDLC)**

In SDLC mode, this bit indicates reception of a valid ending flag and validation of CRC error and residue codes.

D6 **CRC/Framing Error**

If a framing error occurs (in asynchronous modes), this bit is set (and not latched) only for the character on which it occurred. Detection of a framing error adds an additional half bit time to the character time so that the framing error will not also be interpreted as a new start bit. In synchronous modes, this bit indicates the result of comparing the CRC to the appropriate value.

D5 **Receive Overrun Error**

This indicates that more than four characters have been received without a read from the CPU. Only the character that has been written over is flagged with this error, but when this character is read, the error condition is latched until reset by the error reset command (command 6, WR0). If the Status Affects Vector bit is enabled (D2, WR1), the character that has been overrun will interrupt with the special receive condition vector (D3-D1, RR2).

D4 **Parity Error**

When parity is enabled (D0, WR4), this bit is set for those characters whose parity does not match the programmed one (D1, WR4). The bit is latched so that once an error occurs, the bit remains set until the Error Reset command (command 6, WR0) is given.

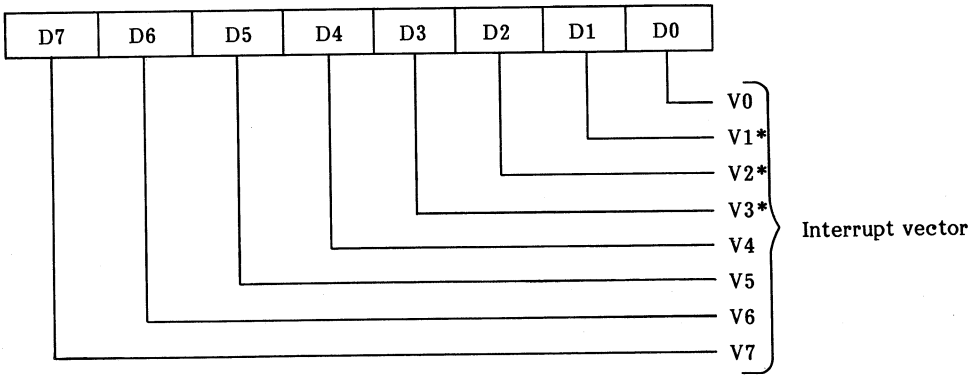
D3-D1 **Residue Codes**

In the SDLC mode, these bits indicate the length of the information field, when the I field is not an integral multiple of the character length. These codes are meaningful upon transfer of the End of Frame bit. If the last character boundary coincides with the boundary of the I-field and CRC field, the residue code will always be 011.

D0 **All Sent**

This bit is set when all characters have completely cleared the transmitter (XMT Buffer Empty) in asynchronous modes. Interrupts are not caused by transitions of this bit. It is always set in synchronous modes.

READ REGISTER 2 (Channel B Only)



* Variable if "Status Affects Vector" is programmed

Figure 3-16. Read Register 2 Bit Map

	V_3	V_2	V_1	
Ch B	0	0	0	Ch B Transmit Buffer Empty
	0	0	1	Ch B External/Status Change
	0	1	0	Ch B Receive Character Available
	0	1	1	Ch B Special Receive Condition*
Ch A	1	0	0	Ch A Transmit Buffer Empty
	1	0	1	Ch A External/Status Change
	1	1	0	Ch A Receive Character Available
	1	1	1	Ch A Special Receive Condition

*Parity Error, RCV Overrun Error, Framing Error, and End of Frame are Special Receive Conditions.

This register contains the interrupt vector as written into WR2 if the Status Affects Vector bit (D2, WR1) is not set. If the Status Affects Vector bit is set, RR2 contains the interrupt vector as it would be returned were an interrupt from the SIO to be processed exactly at the time of the read. When the Status Affects Vector bit is set, status information is programmed into the vector to provide different interrupt handling routines for various SIO conditions. The vector addresses refer to a table in memory which contains the actual starting address location of the interrupt service routine. The vector address from the SIO contains the low order byte of the Z80A interrupt service routine table, with the high order address byte taken from the Z80A I register. If no interrupts are pending, V3 = 0, V2 = 1, V1 = 1 and other bits are as programmed. This register can be read only through channel B.

SIO OPERATION

Initialization

The SIO should be initialized in the following order: Write to the CTC to select a baud rate clock, then initialize the SIO registers. WR2 and WR4 must be initialized before WR1, 3, 5, 6, and 7.

Baud Rate

Selection of baud rates is a function of both the counter/timer and the SIO. Figure 3-17 gives the values to be programmed into the counter/timer and the SIO for all possible standard baud rates. Initializing the counter/timer is covered under the Counter/Timer heading in this section (see page 3-31). Software for initializing the SIO and for selecting a baud rate is provided in section 4 of this manual.

The 7842 is available in three clock speeds, 4.0 MHz (7842-0), 3.6864 MHz (7842-1), and 2.5 MHz (7842-2). The 3.6864 MHz clock, although 8 percent slower than the 4 MHz clock, divides evenly to give accurate standard baud rates. The 2.5 MHz version is for use with slower memory devices. The maximum baud rates achievable with the RS-422 interface are as follows: 7842-0, 4.0 MHz, 500K synchronous, 62.5K asynchronous; 7842-1, 3.686 MHZ, 460.8K synchronous, 57.6K asynchronous; 7842-2, 2.5 MHz, 312.5K synchronous, 38.4K asynchronous (see figure 3-17). The maximum baud rate with the RS-232-C interface is 19.2K, both synchronous and asynchronous.

General Operation

General operation is largely dependent on the specific application and is mostly a function of the use of the registers. The best way to prepare an operating procedure is to review the functions of the registers with your application in mind. Several aspects of operation are covered in the following headings.

Interrupts

The SIO has an internal interrupt priority chain. The priorities within the SIO and their relationship to CTC interrupts are shown in figure 3-18, page 3-29. The most common way to operate the SIO is to use the Status Affects Vector feature and program the Z80A for interrupt mode 2. This is the mode in which the vector is combined with the contents of the Z80A interrupt register to form a 16-bit vector. This vector then points to a location in memory where the address of the interrupt service routine resides. When using this mode, be sure to:

1. Program the Z80A interrupt register. It forms the upper half of the 16-bit vector.

2. Set the least significant bit of the vector programmed into the SIO to 0. This is a requirement of the Z80A.

3. Use the Z80A return from interrupt command at the end of the interrupt service routine. This is a requirement of the SIO.

This interrupt plan must be used if the CTC also generates interrupts. The CTC can only operate in mode 2.

4.000 MHz Z80A

Baud Rate	CTC Clock Divider	CTC Count HEX	CTC Count DEC	CTC MODE	SIO Clock Divider	Actual Baud Rate % Error
500K	1	02	02		1	- 0 -
200K	1	05	05		1	- 0 -
100K	1	0A	10		1	- 0 -
62.5K	1	10	16		1*	- 0 -
50K	1	14	20		1	- 0 -
25K	1	28	40		1	- 0 -
20K	1	32	50	Counter	1*	- 0 -
19.2K	1	34	52		1	+0.16
9600	1	68	104		1	+0.16
4800	1	D0	208		1*	+0.16
2400	1	1A	26		16	+0.16
1800	1	23	35		16	+0.79
1200	1	34	52		16	+0.16
600	1	68	104		16	+0.16
300	16	1A	26		16	+0.16
200	16	27	39		16	+0.16
150	16	34	52		16	+0.16
134.5	16	3A	58	Timer	16	+0.15
110	16	47	71		16	+0.03
75	16	68	104		16	+0.16
50	16	27	39		64	+0.16

***NOTE:** This baud can be achieved both synchronously (SIO Clock Divider = 1) or asynchronously (SIO Clock Divider = 16).

Figure 3-17. Baud Rate Selection (7842)

3.6864 MHz Z80A

Baud Rate	CTC Clock Divider	CTC Count HEX	CTC Count DEC	CTC MODE	SIO Clock Divider	Actual Baud Rate % Error
460.8K	1	02	02		1	- 0 -
307.2K	1	03	03		1	- 0 -
153.6K	1	06	06		1	- 0 -
76.8K	1	0C	12		1	- 0 -
57.6K	1	10	16	Counter	1*	- 0 -
38.4K	1	18	24		1	- 0 -
19.2K	1	30	48		1*	- 0 -
9600	1	60	96		1*	- 0 -
4800	1	C0	192		1*	- 0 -
2400	1	18	24		16	- 0 -
1800	1	20	32		16	- 0 -
1200	1	30	48		16	- 0 -
600	1	60	96		16	- 0 -
300	16	18	24		16	- 0 -
200	16	24	36		16	- 0 -
150	16	30	48	Timer	16	- 0 -
134.5	16	36	54		16	-0.87
110	16	41	65		16	+0.07
75	16	60	96		16	- 0 -
50	16	24	36		64	- 0 -

***NOTE:** This baud can be achieved both synchronously (SIO Clock Divider = 1) or asynchronously (SIO Clock Divider = 16).

Figure 3-17. (Continued)

2.5 MHz Z80A

Baud Rate	CTC Clock Divider	CTC Count HEX	CTC Count DEC	CTC MODE	SIO Clock Divider	Actual Baud Rate % Error
312.5K	1	02	02		1	- 0 -
125.0K	1	05	05		1	- 0 -
62.5K	1	0A	10		1	- 0 -
38.4K	1	10	16		1*	+1.72
19.2K	1	21	33		1*	-1.36
9600	1	41	65		1*	+0.16
4800	1	82	130		1*	+0.16
2400	1	10	16	Counter	16	+1.72
1800	1	16	22		16	-1.36
1200	1	21	33		16	-1.36
600	1	41	65		16	+0.16
300	16	10	16		16	+1.72
200	16	18	24		16	+1.72
150	16	21	33		16	+1.36
134.5	16	24	36	Timer	16	+0.84
110	16	2C	44		16	+0.38
75	16	41	65		16	+0.16
50	16	62	98		16	-0.35

***NOTE:** This baud can be achieved both synchronously (SIO Clock Divider = 1) or asynchronously (SIO Clock Divider = 16).

Figure 3-17. (Continued)

PRIORITY	DESCRIPTION	VECTOR
Highest		
1	CTC Channel 0	XXXX X000
2	CTC Channel 1	XXXX X010
3	CTC Channel 2	XXXX X100
4	CTC Channel 3	XXXX X110
5	SIO Ch A, RCV character or special RCV condition	XXXX 110X XXXX 111X
6	XMT Buffer Empty	XXXX 100X
7	SIO Ch B, RCV character	XXXX 010X
8	SIO Ch B, RCV character or special RCV condition	XXXX 010X XXXX 011X
9	XMT Buffer Empty	XXXX 000X
10	Handshake Transition	XXXX 001X
Lowest		

Figure 3-18. Interrupt Vectors and Priority (7842)

If properly selected, the vectors for the CTC and SIO are consecutive (2 bytes apart). If the CTC vectors are to precede the SIO vectors, program the CTC with a vector of $X8_H$ and the SIO with a vector of $(X+1)0_H$. The vectors run from $X8_H$ to $(X+1)E_H$. If the SIO vectors come first, program the SIO with a vector of $X0_H$ and the CTC with a vector of $(X+1)0_H$. The vectors run from $X0_H$ to $(X+1)6_H$.

There are three circumstances under which the interrupts should be handled differently than previously explained.

1. Interrupts are not used at all.

   Monitor the interrupt pending bit of RR0. When set, read the vector from RR2 (having previously initialized the Status Affects Vector mode). Use the vector to determine the appropriate response and use the return from interrupt command of WR0 after servicing the SIO. The return from interrupt command re-enables the SIO priority chain.

2. Other cards in the system generate interrupts but only provide 8080-type restart vectors.

   Use Z80A interrupt mode 0 and program the SIO with a restart instruction for a vector. Do not use the Status Affects Vector mode. Use the Z80A return from interrupt command after servicing the SIO.

3. Other cards in the system generate interrupts, but do not provide vectors.

   Use Z80A interrupt mode 1. Upon interrupt, check the various possible sources of the interrupt. This can be done on the SIO by checking the Interrupt Pending bit of RR0. If the SIO has generated the interrupt, check the XMT Buffer Empty and RCV Character Available flags. If neither of these is set, check the handshake and break flags. All of these are in RR0. If the RCV Character Available flag is set, check the error flags of RR1 to see if there were any errors in the reception. Use the Z80A return from interrupt command or the return from interrupt command of WR0 after servicing the SIO. This re-enables the priority chain within the SIO.

   The Status Affects Vector mode and the vector in RR2 cannot be used in this mode. The modified vector is only available while the interrupt is pending and will already have returned to its initial condition before the program can check it.

RS-232-C Handshaking

Three elements control handshaking for the RS-232-C port: the SIO, the processor, and the jumpers. The jumpers (figure 2-16, page 2-17) select to which RS-232-C line each handshake pin of the SIO is connected. The SIO can control some of the handshake lines, and the processor can control all or some of them through the SIO registers, depending upon whether or not the auto handshaking is enabled.

Figure 3-19 shows which lines can be controlled by the SIO. The CTS* and DCD* pins automatically enable/disable the transmitter and receiver if auto handshaking is selected. If the receiver or transmitter are disabled while receiving/transmitting a character, the character may be lost. It is recommended that the character be repeated. Perform a channel reset command through WR0, reinitialize, and repeat the character.

If auto handshaking is not enabled, the CTS* and DCD* pins must be monitored by the processor or they can generate interrupts if handshake interrupts are enabled. CTS* and DCD* are monitored through RR0.

The DTR* and RTS* pins are controlled by the processor through WR5. There are no handshaking lines on the RS-422 interfaces.

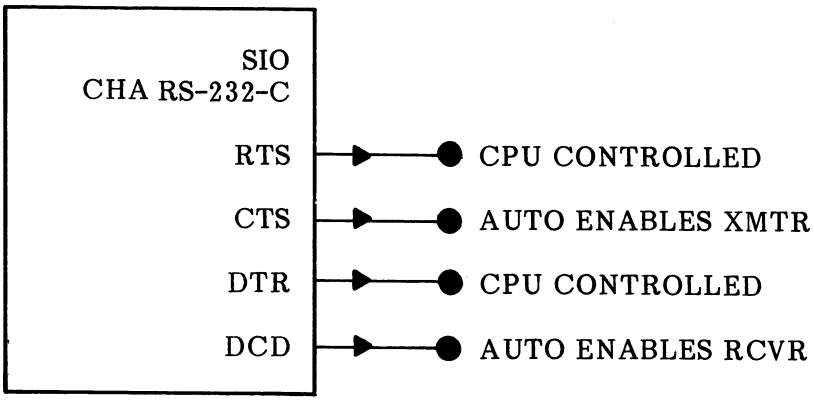


Figure 3-19. SIO RS-232-C Handshake Capability (7842)

Block Transfers

Block transfers can occur between the SIO and the CPU or a DMA controller. Block transfers are controlled by the two wait/ready lines (W/RDY A* and W/RDY B*), one for each channel.

For DMA block transfers, the wait/ready line should be programmed in the ready mode (see the description of bits 5, 6, and 7 of WR1, pages 3-9 and 3-10). In this mode, the line indicates to the DMA controller when the SIO is ready to transfer another byte.

For CPU block transfers, the wait/ready line should be programmed in the wait mode. The line is connected to the STD WAITRQ* line through jumper W8 (see description in section 2, page 2-18). The wait/ready line can be enabled at initialization and left enabled if all transfers will be block transfers. Since the line only becomes active when the processor attempts to read or write a character before the SIO is ready, it does not interefere with normal processor operation. Reading from or writing to the control or status registers is not affected by the wait/ready line.

The CPU block transfers are performed using the Z80A block I/O transfer instructions. Blocks of up to 256 bytes can be transferred. The Interrupt on First Character mode should be used when receiving in the block mode. In the Wait mode, the wait/ready line is open active, so a pull-up resistor must be used.

COUNTER/TIMER

The Counter/Timer Chip (CTC) can be used for a number purposes including timing (based on the onboard clock), counting (of external events), inputting external interrupts (to prioritize them and generate unique vectors), generating the baud rate clock for the SIO, and generating timed output pulses. The counter/timer circuit has four independent counter/timers (channels) that are programmed through individual channel control registers and time constant register, and a common interrupt vector register.

All four channels can generate an interrupt when they reach a zero count. Three of the channels also have an output line that indicates when a zero count has been reached. One of these channels has its output line dedicated to baud rate generation, and another can be used as an output or for baud rate generation. The channel function uses are shown in figure 3-20 as the CTC is implemented on the 7842.

	CHANNELS WHICH CAN BE USED FOR EACH FUNCTION		
FUNCTION	NO BAUD CLOCKS	1 BAUD CLOCK	2 BAUD CLOCKS
External Inputs (for counting or Interrupt inputs)	1,2,3	1,2,3	2,3
Timers	0,1,2,3	1,2,3	2,3
Output Signals	1,2	1,2	2

Figure 3-20. CTC Channels per Function

The interrupt vector register is coded by the CTC to indicate the interrupting channel and to provide a unique vector for each channel. This vector is used by the Z80A in mode 2 operation. Each channel can be used with the system clock (timer mode) or an external clock (counter mode with a separate clock input per channel). The CTC input to the SIO must be equal to or less than one-fifth the rate of the system clock, Φ, used by the SIO. The SIO also requires minimum high and low pulse magnitudes of 180 nanoseconds from the CTC.

The internal structure of the CTC is shown in figure 3-21. Each channel has a control register, an initial count or "load" register, and a current count register used to read the counter. Channel 0 also has a vector register. The control and current count registers are directly accessible. The load registers and channel 0 vector register are addressed through the control registers.

CTC Counter Mode

In the counter mode, the down counter is decremented from the value loaded into it from the time constant register. The channel down counter is decremented by the CLK/TRG input for the channel. The ZC/TO output is pulsed active high for a zero count and the count begins again. Channel 3 has no output pin, and can only interrupt. The CLK/TRG input pulse must have a minimum width of 120 nanoseconds, and the minimum period must not be less than twice the system clock period.

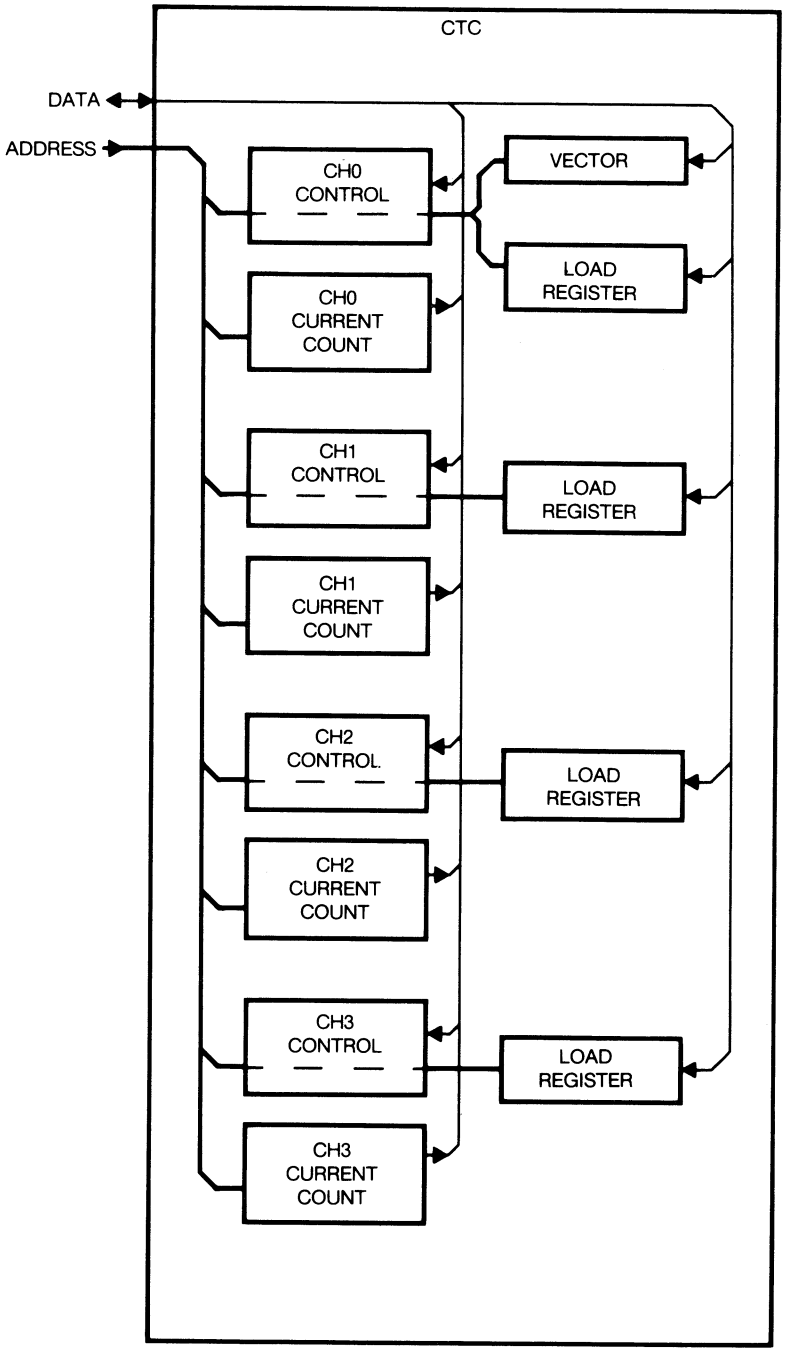


Figure 3-21. CTC Internal Structure

The counter mode has two uses: external event counter and interrupt controller. As a counter, the CTC can simply be used as an 8-bit down counter. To obtain a larger counter, the ZC/TO output of one channel can be cascaded into the CLK/TRG input of another channel. This is explained further under Cascading on page 3-38.

When used as a baud rate generator, the pulse output of the CTC is shaped by a flip-flop. This divides the CTC baud rate output by two, and gives the TXCLK the 50% duty cycle required for synchronous serial communications. The baud rate with the CLK/TRG input connected to (Z80A Φ)/2 equals (Z80A Φ)/2 times ½ (flip-flop waveshaping) times the CTC prescale (1 for counter mode) times 1/n (where n is the time constant value, an integer from 1 to 256) times the SIO chip scale (1, 1/16, 1/32, or 1/64):

$$\text{Baud rate} = \frac{\text{Z80A } \Phi}{2} \times 1/2 \times \text{CTC scale (1)} \times 1/n \times \text{SIO scale}$$

As an interrupt controller, the CTC can be used to provide prioritized, independently vectored, maskable, edge selectable, count programmable, external interrupt inputs. The counter is set to one, and an active edge at the CLK/TRG input triggers a down count to zero, and causes an interrupt (if enabled). A second interrupt on the same channel cannot interrupt until after the return from interrupt (RTI) instruction of the interrupt service routine of the first interrupt.

CTC Timer Mode

In the timer mode, the system clock clocks a prescaler that divides the system clock by 16 or 256 (scale is selected by the channel control word). The output of the prescaler decrements the down counter. Thus the timer mode measures intervals of time based on the system (CPU) clock. In the timer mode, an output occurs at the ZC/TO channel output at a zero count and the operation begins again. The ZC/TO output pulse occurs at a period given by: the system clock period times the prescaler factor (16 or 256) times the programmed time constant. The timer can be selected by the channel control register to begin almost immediately after loading, or after a trigger at the channel CLK/TRG input.

The timer mode can be used for baud rate generation (the CTC prescale is 1/16 or 1/256 for timer mode). The baud rate equals the CPU clock frequency times the CTC prescale (1/16 or 1/256) times 1/n (where n is the time constant value an integer from 1 to 256) times SIO chip scale (1, 1/16, 1/32, or 1/64).

baud rate = Z80A Φ x 1/2 x CTC scale x 1/n x SIO scale

CTC Control Registers

The control register is the same for each channel and is shown in figure 3-22.

D7	D6	D5*	D4	D3*	D2	D1	D0
ENABLE INTS	COUNT-ER/TIMER	CLOCK DIVIDER	CLOCK EDGE	TRIGGER	LOAD REG.	CHANNEL RESET	LOAD VECTOR

BIT	DESCRIPTION
7	1 = Enable Interrupts
6	1 = Counter Mode 0 = Timer Mode
5*	1 = Clock divided by 256 0 = Clock divided by 16
4	1 = External clock/trigger line is rising edge sensitive 0 = Falling edge sensitive
3*	1 = Timer started by external trigger 0 = Timer starts after being loaded
2	1 = Next byte to this channel is load register data
1	1 = Channel stops; must be reloaded or re-initialized Momentary function
0	Channel 0 only; 0 = byte to be loaded to vector register

*Timer mode only

Figure 3-22. CTC Control Registers (7842)

D0 Load Vector

Bit 0 must always be set to a 1 when writing to a control register, except when writing the vector. If the bit is set to a 0 when writing to the channel 0 control register, the byte is loaded directly into the vector register. It has no effect on the control register. A byte written to any other control register with bit 0 equal to 0 has an undefined effect.

D1 Channel Reset

This bit stops the counter. It does not need to be reset to continue operation. The channel resumes operation as soon as a new control and/or load register byte are written. If both this bit and the load register bit are set, the channel resumes operation as soon as a new load register byte is written. This bit does not clear the control register and the other bits are affected when writing to this bit. Therefore, be sure to write the other bits to the desired state.

D2 Load Register

This bit sets up the CTC channel to accept the next byte written to that channel as load register data.

D3 Trigger

This bit is used only in the timer mode. It enables the timer to be started by a pulse on the external input for that channel. Otherwise, the timer starts as soon as the count value is loaded.

D4 Clock Edge

With this bit set, the external input is positive edge sensitive. Otherwise, it is negative edge sensitive.

D5 Clock Divider

This bit is used in the timer mode only. If set, the main CTC clock input is divided by 256 and then drives the timer. If the bit is reset, the clock is divided by 16.

D6 Counter/Timer

If set, the counter mode is selected. If reset, the timer mode is selected.

D7 Enable Interrupts

If set, this allows an interrupt to be generated whenever the down counter reaches 0.

Load Register

The load register is accessed through the control register. A value must be entered in the load register before the channel will operate. In the timer mode, if the timer is not started by an external trigger, it starts immediately after a value is written to the load register.

The counter is reloaded from the load register as soon as it reaches a count of zero. If a new value is loaded while the channel is operating, the channel will finish its current count before being loaded with the new count. The lowest value is 01, the highest is 00, which yields 256 counts.

Current Count Register

The current count register makes available the value in the counter. It is latching so that the value read is always valid. It is not possible to catch the counter in transition and get an invalid reading.

Since the counter is a down counter only, an inversion is required when counting external events or measuring time intervals. Load the channel with FF_H initially, then invert the number read from the current count register to get the actual count.

Vector Register

The vector register only exists in channel 0. It is accessed by writing to the control register with bit 0 cleared. The byte thus written is loaded directly into the vector register. Bit 0 must always be zero (the Z80A requires this) and bits 1 and 2 are modified by the CTC to provide a unique vector for each channel. Only bits 3 through 7 are selected by the user.

CTC OPERATION

To operate the CTC, write to the control register and load register. If interrupts are used, the vector register must also be initialized. The CTC interrupts can only be used in Z80A interrupt mode 2.

Interrupts: The internal interrupt priority of the CTC and the vectors generated by each channel are shown in figure 3-18.

The CTC can be used as an external interrupt prioritizer. Use the CTC in the counter mode and place a count of one in it. Every time an active edge is received on the external input line, the count goes to zero and an interrupt is generated.

Care must be taken when disabling the CTC interrupts. It is possible to disable the CTC interrupts just as an interrupt is being generated. In this case, the CTC may not provide the interrupt vector when it is requested by the processor. For this reason, it is recommended that the Z80A interrupts be temporarily disabled just before disabling the CTC interrupts, then re-enabled.

The Z80A return from interrupt command should always be used at the end of an interrupt service routine. This re-enables the interrupt priority chain.

Cascading: Channels can be cascaded together to give higher counts or longer time delays. To cascade, place one channel in the counter mode and jumper its external input to the output signal of another channel at connector J2. This other channel may be in the counter or timer mode. Each time it reaches a count of zero, it decrements the channel whose input it is driving.

Output Pulses: The CTC can be used to generate a series of pulses since the output line goes low momentarily each time the counter reaches zero. The frequency is a function of the count value and the clock divider.

Software Count/Trigger: Once a channel has been initialized, it can be decremented or triggered by software. In the counter mode, the processor can decrement the counter; in the timer mode, the processor can act as a trigger to start the timer. Write a new control register byte with the clock edge bit complemented. Once a channel has been initialized, any change in this bit besides changing the polarity of the external input, also acts as a pulse on the input. This can be used to decrement a counter or start a timer.

CTC Timing: Detailed timing for the CTC can be found in data sheets on Zilog's Z8430A and Mostek's MK3882-4. Listed here are the main points about the CTC timing which should be considered before operation.

- An additional cycle is not required for the zero count. As an example, if loaded with 4 the down count would be 4, 3, 2, 1, 4, 3, ...
- The output line goes low at zero count, but does not stay low. It returns high after approximately one clock cycle, even in the counter mode.
- The incoming count pulses must not be more than half the CTC clock rate (4.0 MHz for 7842-0, 3.6864 MHz for 7842-1, 2.5 MHz for 7842-2).
- When a count pulse is received, the down counter decrements on the next rising edge of the CTC clock. If the count pulse occurs too close to the rising edge of the clock, the decrement occurs on the next rising edge of the clock.
- The timer starts or is ready for a trigger pulse after the machine cycle of the instruction which writes to the CTC. What is written may be a new control word that does not specify a load register word to follow or a load register word. Specifically, the timer is enabled on the rising edge of the second clock plulse after the machine cycle.
- If the timer is started by external trigger, it starts on the second rising clock edge after the trigger. If the trigger occurred too close to a rising edge, the timer starts after the third rising clock edge.
- If a control word is to be followed by a load register word, the timer trigger is not enabled until the load register word is written.
- External inputs must remain high or low at least 200 nanoseconds.

INTRODUCTION

This section contains hardware-level subroutine modules with which to operate the 7842 card. The software in this section can be used without license from Pro-Log. Although tested and believed correct, this software is not represented to be free from errors or copyright infringement, or appropriate for any specific application.

The subroutines are in STD instruction mnemonics. The coding forms are grouped at the end of this section, following the flowcharts and subroutine specifications.

MEMORY ADDRESSES

Full memory addresses are given. They are preferred addresses that allow the subroutines to work with those provided for other Series 7000 STD BUS cards from Pro-Log. The program addresses correspond to the 7842 onboard memory sockets.

If your system cannot use the memory addresses in the 7842 software package, simply change the memory page addresses, as required, when loading these modules into your system.

I/O PORT ADDRESSES

The 7842 I/O ports are assigned preferred hexadecimal addresses F0-F7 and FE for compatibility with other Series 7000 cards. Section 2 shows how to remap these addresses if necessary. This software can be used by simply changing the port addresses when loading the program modules into your system.

DESCRIPTION

The software in this section can be used to operate both the SIO and CTC. It is written mainly for the SIO, but the subroutines which initialize the CTC for baud rate generation may be used for general purpose CTC control. The software provided in this manual is written to display the various functions of the SIO. Some jumpers may need to be moved depending upon which serial communications format and interface is used. Using this software, the SIO will transmit and receive asynchronously at 4800 baud on channel A, and synchronously at 19.2K baud, using SDLC format, on channel B.

There are 16 subroutines, most of which are subroutines within subroutines and are not accessed directly. Only seven of them need to be accessed directly to perform serial communications. Others are useful as building blocks in the user's own software for serial communications or CTC control. All of these are shown in figure 4-1.

This software is not complete in itself. There is no software provided to process errors in data reception or changes in incoming handshake lines, since these are largely application dependent. If used as a package with user-supplied software, these subroutines can greatly simplify the use of the serial ports.

NAME	DESCRIPTION	MAIN SUB-ROUTINE	MAY BE USED BY ITSELF	NOT USEFUL BY ITSELF
(INIT COMM)	Initialize Communications Interface	X		
(INIT INTER)	Initialize Z80 Interrupt		X	
(INIT CTC 0)	Initialize CTC Channel 0		X	
(INIT CTC 1)	Initialize CTC Channel 1		X	
(INIT SIO)	Initialize entire SIO		X	
(INIT SIO CH)	Initialize one channel of SIO		X	
(XMT CH A ASYN)	Transmit asynchronously a block of data (Ch. A)	X		
(XMT CH B SDLC)	Transmit a block of data in SDLC format (Ch. B)	X		
(XMT BLK B)	Transmit a block of data (Ch. B)			X
(END XMT A?)	Verify that entire block has been transmitted (Ch. A)			X
(END XMT B?)	See above (Ch. B)			X
(XMT CHAR)	Transmit a character			X
(RCV CH A ASYN)	Receive a character (Ch. A)	X		
(RCV CH B SDLC)	Receive a character (Ch. B)	X		
(SPCL RCVA ASYN)	Determine error (Ch. A)	X		
(SPCL RCVB SDLC)	Determine error/EOF (Ch. B)	X		

Figure 4-1. Subroutine Types (7842)

If the software provided is used as a package, only a few requirements must be met by the user. The seven main subroutines and their requirements are given in figure 4-2. The following list details the requirements, and the points to be considered before implementing the software.

- No software is provided for handshake control or for responding to changes in incoming handshaking.

- No software is provided for processing of errors in received data.

- No software is provided to respond to received data. The character is simply received and stored. Jump to Subroutine instructions are provided to jump to the user's own subroutines. The user's subroutine may then respond to the character according to the requirements of the application.

- This software operates with Z80A interrupt mode 2. It selects mode 2 and enables the interrupt as part of the initialization procedure. It also loads the Z80A interrupt register. The data to be loaded into the register must be selected by the user (see address 1D1B).

- The baud rates and various other factors selected by the initialization routine should be checked to see that they match the application. The initialized values are given in the description of the (INIT COMM) routine.

- The number of bytes to be transferred must be in register A (FF = 255) and the address of the data to be transferred in the HL register pair before using the transmit routines.

- Keep in mind the 7842 onboard interrupt priority (figure 3-18, page 3-29).

NAME	DESCRIPTION	USE REQUIREMENTS
(INIT COMM)	Initialize Communications Interface (Entry Address 1D00)	None
(XMT CH A ASYN)	Transmit a block of data asynchronously (Ch. A) (Entry address 1D95)	Data in memory, data pointer in register pair HL, number of bytes in register A
(XMT CH B SDLC)	Transmit a block of data in SDLC format (Ch. B) (Entry Address 1DC0)	Control and data in memory, data pointer in register pointer in register pair HL, number of bytes in register A
(RCV CH A ASYN)	Receive and store a character (Ch. A) (Entry Address 1E40)	User provides software to process received data
(RCV CH B SDLC)	Receive and store a character (Ch. B) (Entry Address 1E50)	
(SPCL RCVA ASYN)	Determine asynchronous transmission error (Ch. A) (Entry Address 1E5B)	User provides software to process errors
(SPCL RCVB SDLC)	Determine SDLC transmission error or end of frame (Ch. B) (Entry Address 1E00)	User provides software to process errors and data frame.

Figure 4-2. Main Subroutines Use Requirements (7842)

SUBROUTINE SPECIFICATIONS (INIT COMM)

ENTRY ADDRESS	:	1D00
MEMORY REQUIREMENTS		
Program Memory	:	21 (102 with subroutines and tables)
RAM	:	0
Stack	:	4
REGISTERS ALTERED	:	A, F, B, C, H, L
SUBROUTINES USED	:	(INIT CTC 0), (INIT CTC 1), (INIT SIO), (INIT INTER)
EXECUTION TIME	:	1445 Time States
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION	:	

This subroutine initializes the SIO, the CTC (to generate baud rates), and the Z80A interrupt. The initialized conditions should be checked to be sure they match the application. The initial baud rates are 4800 for channel A and 19.2K for channel B using the 4 MHz version (7842-0).

SUBROUTINE SPECIFICATIONS (INIT INTER)

ENTRY ADDRESS	:	1D18
MEMORY REQUIREMENTS		
Program Memory	:	8
RAM	:	0
Stack	:	0
REGISTERS ALTERED	:	A, I
SUBROUTINES USED	:	None
EXECUTION TIME	:	38 Time States
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION	:	

This subroutine selects Z80A interrupt mode 2 and enables the Z80A to interrupt. It also loads the Z80A interrupt register. The byte to be loaded in the interrupt registers must be provided by the user and stored at address 1D18.

SUBROUTINE SPECIFICATIONS: (INIT CTC 0)

ENTRY ADDRESS	:	1D20
MEMORY REQUIREMENTS		
Program Memory	:	9
RAM	:	0
Stack	:	0
REGISTERS ALTERED	:	A
SUBROUTINES USED	:	None
EXECUTION TIME	:	46 Time States
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION:	:	

This subroutine initializes CTC channel 0 to 4800 baud.

SUBROUTINE SPECIFICATIONS: (INIT CTC 1)

ENTRY ADDRESS	:	1D30
MEMORY REQUIREMENTS		
Program Memory	:	9
RAM	:	0
Stack	:	0
REGISTERS ALTERED	:	A
SUBROUTINES USED	:	None
EXECUTION TIME	:	46 Time States
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION:	:	

This module initializes CTC channel 1 to 19.2K baud.

SUBROUTINE SPECIFICATIONS: (INIT SIO)

ENTRY ADDRESS	:	1D40
MEMORY REQUIREMENTS		
Program Memory	:	13 (55 with subroutine and table)
RAM	:	0
Stack	:	2
REGISTERS ALTERED	:	A, F, B, C, H, L
SUBROUTINES USED	:	(INIT SIO CH)
EXECUTION TIME	:	1204 Time States maximum
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION:	:	

This subroutine initializes both channels of the SIO: channel A for asynchronous communications and channel B for SDLC communications.

SUBROUTINE SPECIFICATIONS: (INIT SIO CH)

ENTRY ADDRESS	:	1D50
MEMORY REQUIREMENTS		
Program Memory	:	12 (plus table for one or two channels)
RAM	:	0
Stack	:	0
REGISTERS ALTERED	:	A, B, C, H, L, F
SUBROUTINES USED	:	None
EXECUTION TIME	:	693 Time States maximum
ENTRY REQUIREMENTS		
Register Pair HL	:	Points to CH A ASY 232 data table (1D60), CH A ASY 422 data table (1D70), or CH B SDLC data table (1D80)
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION:	:	

This subroutine initializes one channel of the SIO. The channel is selected by loading register pair HL with the appropriate address (see Entry Requirements). If mode 2 interrupts are used, write register 2B must contain the interrupt vector for both SIO channels.

SUBROUTINE SPECIFICATIONS: (XMT CH A ASYN)

ENTRY ADDRESS:	:	1D95
MEMORY REQUIREMENTS		
Program Memory	:	37 (85 with subroutines)
RAM	:	3
Stack	:	10
REGISTERS ALTERED:	:	None
SUBROUTINES USED	:	(END XMT A?), (XMT CHAR)
EXECUTION TIME:	:	489 Time States + 252 Time States per additional character (if transmission of previous block has been completed).
ENTRY REQUIREMENTS		
Register A	:	Number of bytes to be sent (FF=255)
Register Pair HL	:	Address of first byte to be sent
Memory	:	Data to be sent
EXIT CONDITIONS	:	Byte count decremented and stored at 3200, data pointer incremented and stored at 3201 and 3202.
FUNCTIONAL DESCRIPTION	:	

This subroutine asynchronously transmits a block of data of up to 255 bytes. The program first checks to see if a previous block is still being sent. If it is, the program stays in a continuous checking loop. When the block can be sent, the first byte is sent and the program exits. Thereafter, whenever the transmitter indicates it is ready, via interrupt, the program responds by sending another byte. When the transmitter interrupts after sending the last byte, the program turns off the transmitter interrupt.

SUBROUTINE SPECIFICATIONS: (XMT CH B SDLC)

ENTRY ADDRESS	:	1DC0
MEMORY REQUIREMENTS		
Program Memory	:	49 (97 with subroutines)
RAM	:	3
Stack	:	10
REGISTERS ALTERED	:	None
SUBROUTINES USED	:	(END XMT B?) (XMT BLK B), (XMT CHAR)
EXECUTION TIME	:	488 Time States + 252 Time States per additional character (if transmission of previous block has been completed).
ENTRY REQUIREMENTS		
Register A	:	Number of bytes to be sent (FF=255)
Register Pair HL	:	Address of first byte to be sent
Memory	:	Control byte and data to be sent
EXIT CONDITIONS	:	Byte count decremented and stored at 3203, data pointer incremented and stored at 3204 and 3205.
FUNCTIONAL DESCRIPTION	:	

This subroutine transmits a block of data using the SDLC format. The program first checks to see if a previous block is still being sent. If it is, the program stays in a continuous checking loop. When the block can be sent, the transmit CRC generator is reset, the secondary address (see figure 1-4, page 1-6) is sent, and the reset overrun/end of message latch is reset. Thereafter, whenever the transmitter indicates it is ready, via the interrupt, the program responds by sending another byte from memory. When the transmitter interrupts after sending the last byte, the program turns off the transmit buffer interrupt.

SUBROUTINE SPECIFICATIONS: (RCV CH A ASYN)

ENTRY ADDRESS	:	1E40
MEMORY REQUIREMENTS		
Program Memory	:	14
RAM	:	0
Stack	:	4
REGISTERS ALTERED	:	None
SUBROUTINES USED	:	(PROCESS DATA) (user supplied)
EXECUTION TIME	:	80 Time States per character
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION	:	

This is an interrupt service routine that receives an asynchronously transmitted character and stores it in memory. The user must supply a program (PROCESS DATA) to interpret the received data string. The receivers are enabled at initialization, and since reception is interrupt driven, the (PROCESS DATA) routine must flag the processor that a complete message has been received. If a framing error, receiver overrun, or parity error occurs, a Special Receive interrupt is generated.

SUBROUTINE SPECIFICATIONS: (RCV CH B SDLC)

ENTRY ADDRESS	:	1E50
MEMORY REQUIREMENTS		
Program Memory	:	63
RAM	:	0
Stack	:	2
REGISTERS ALTERED	:	None
SUBROUTINES USED	:	(PROCESS CHAR) (user supplied)
EXECUTION TIME	:	63 time States per character
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	N/A
FUNCTIONAL DESCRIPTION	:	

This is an interrupt service routine that receives and stores synchronously transmitted SDLC frames (see figure 1-4, page 1-6). The receivers are enabled at initialization. If errors are detected (SDLC CRC errors or receiver overrun), a Special Receive interrupt is generated. When the final flag is received signaling the end of the frame, a Special Receive interrupt occurs. A subroutine must be supplied to process the frame.

SUBROUTINE SPECIFICATIONS: (SPCL RCVA ASYN)

ENTRY ADDRESS	:	1E5B
MEMORY REQUIREMENTS		
Program Memory	:	37
RAM	:	1
Stack	:	6
REGISTERS ALTERED	:	None
SUBROUTINES USED	:	(FRAMING ERROR), (RECEIVER OVERRUN ERROR), PARITY ERROR) (all user supplied)
EXECUTION TIME:	:	208 Time States
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	Channel A RR1 status stored at 3206
FUNCTIONAL DESCRIPTION	:	

This is an interrupt service routine that indicates transmission errors. If a framing error, receiver overrun, or parity error occurs, the SIO generates a Special Receive vectored interrupt. The user must provide subroutines to process the detected error. After the error is processed, the error code is reset.

SUBROUTINE SPECIFICATIONS: (SPCL RCVB SDLC)

ENTRY ADDRESS	:	1E80
MEMORY REQUIREMENTS		
Program Memory	:	37
RAM	:	1
Stack	:	6
REGISTERS ALTERED	:	None
SUBROUTINES USED	:	(SDLC CRC ERROR), (RCV OVRN CHB), (END OF FRAME B) (all user supplied)
EXECUTION TIME	:	208 Time States
ENTRY REQUIREMENTS	:	None
EXIT CONDITIONS	:	Channel B RR1 status stored at 3207
FUNCTIONAL DESCRIPTION	:	

This is an interrupt service routine that is activated when an SDLC CRC error, receiver overrun, or end of frame is detected. The user must supply subroutines to process the errors and interpret the data received. After processing, the error code is reset.

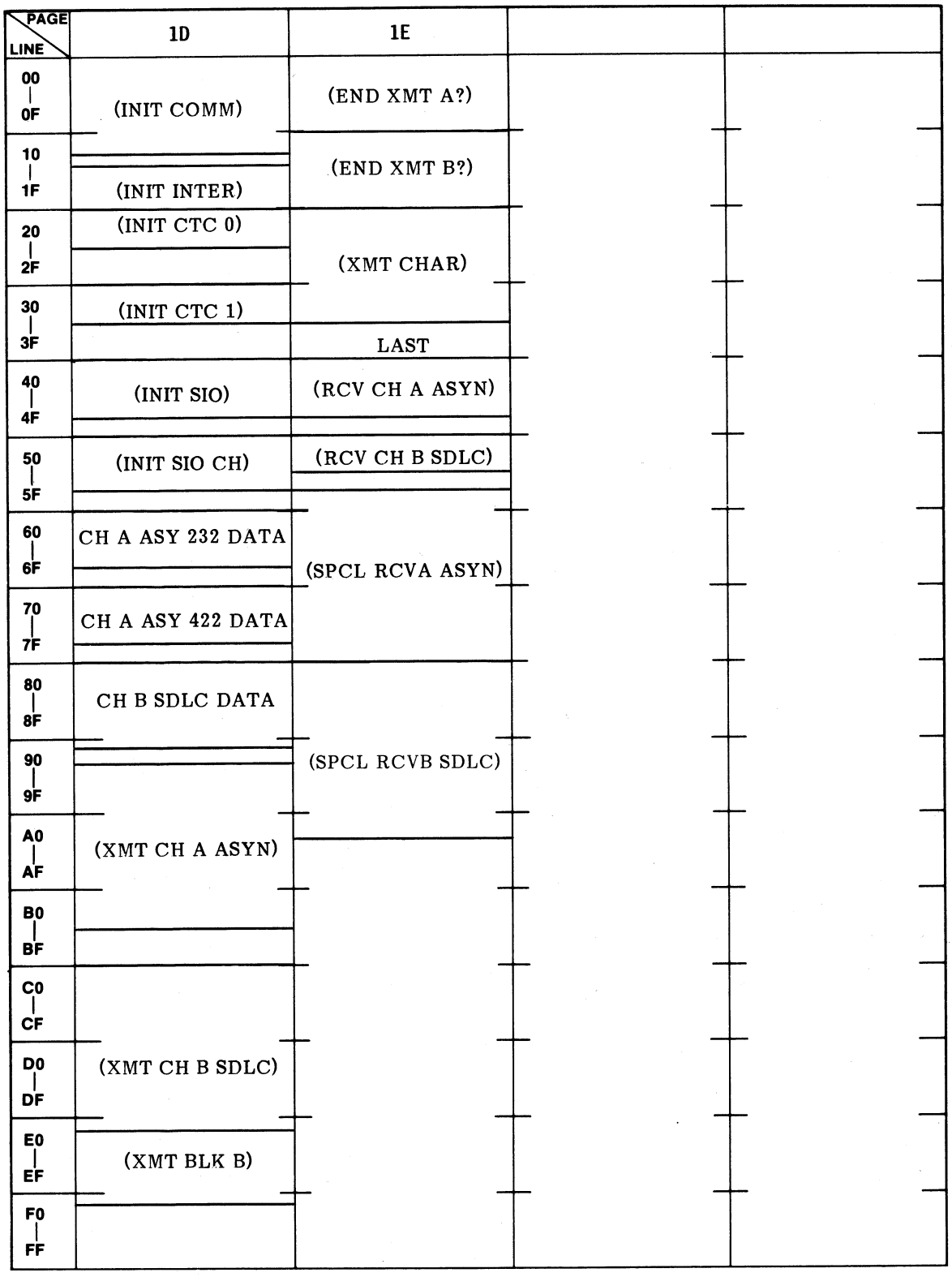
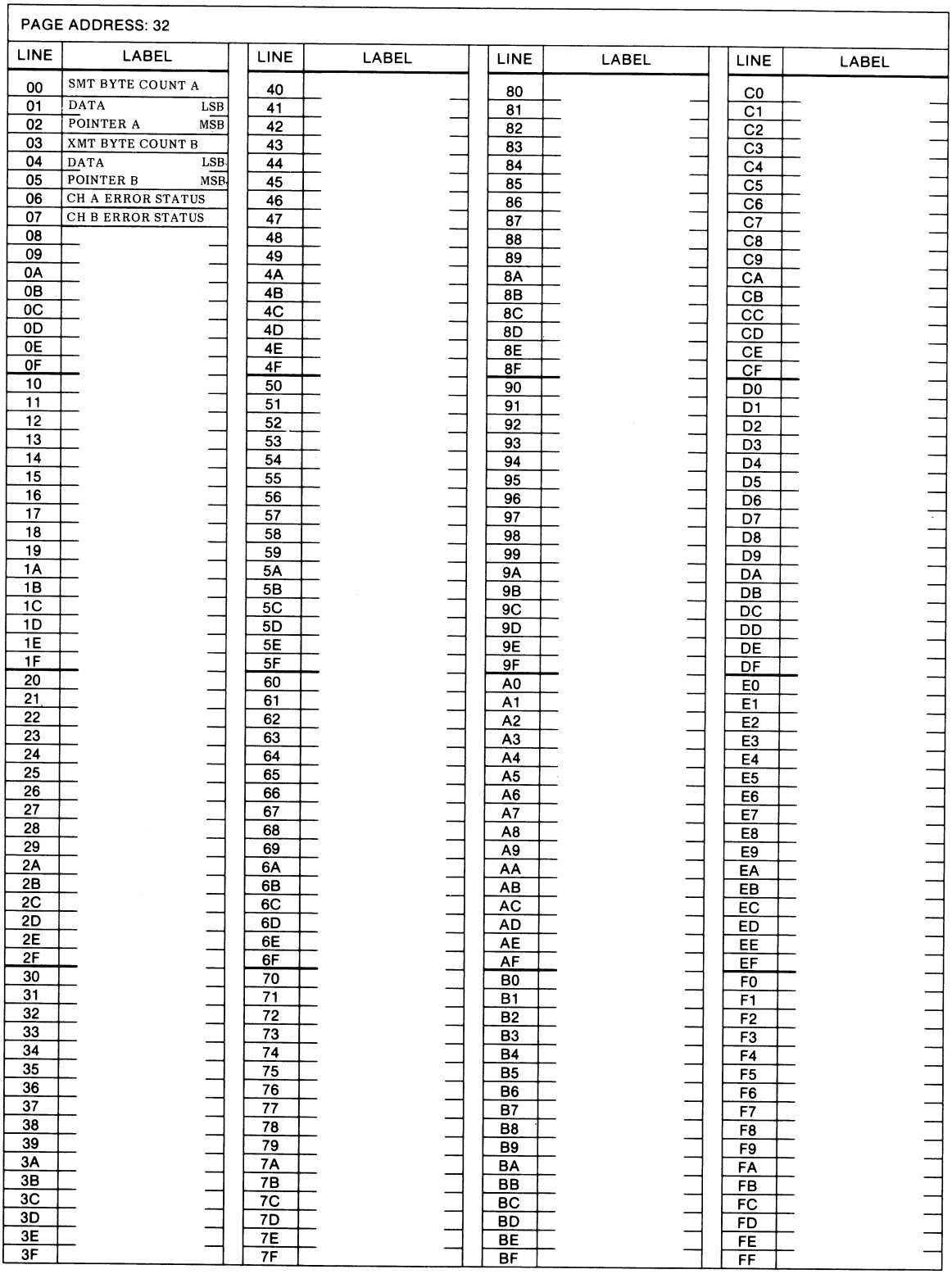


Figure 4-3. 1K Program Memory Map (7842)



PAGE ADDRESS: 32

LINE	LABEL
00	SMT BYTE COUNT A
01	DATA LSB
02	POINTER A MSB
03	XMT BYTE COUNT B
04	DATA LSB
05	POINTER B MSB
06	CH A ERROR STATUS
07	CH B ERROR STATUS
08	
09	
0A	
0B	
0C	
0D	
0E	
0F	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
1A	
1B	
1C	
1D	
1E	
1F	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
2A	
2B	
2C	
2D	
2E	
2F	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
3A	
3B	
3C	
3D	
3E	
3F	

LINE	LABEL
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
4A	
4B	
4C	
4D	
4E	
4F	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
5A	
5B	
5C	
5D	
5E	
5F	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
6A	
6B	
6C	
6D	
6E	
6F	
70	
71	
72	
73	
74	
75	
76	
77	
78	
79	
7A	
7B	
7C	
7D	
7E	
7F	

LINE	LABEL
80	
81	
82	
83	
84	
85	
86	
87	
88	
89	
8A	
8B	
8C	
8D	
8E	
8F	
90	
91	
92	
93	
94	
95	
96	
97	
98	
99	
9A	
9B	
9C	
9D	
9E	
9F	
A0	
A1	
A2	
A3	
A4	
A5	
A6	
A7	
A8	
A9	
AA	
AB	
AC	
AD	
AE	
AF	
B0	
B1	
B2	
B3	
B4	
B5	
B6	
B7	
B8	
B9	
BA	
BB	
BC	
BD	
BE	
BF	

LINE	LABEL
C0	
C1	
C2	
C3	
C4	
C5	
C6	
C7	
C8	
C9	
CA	
CB	
CC	
CD	
CE	
CF	
D0	
D1	
D2	
D3	
D4	
D5	
D6	
D7	
D8	
D9	
DA	
DB	
DC	
DD	
DE	
DF	
E0	
E1	
E2	
E3	
E4	
E5	
E6	
E7	
E8	
E9	
EA	
EB	
EC	
ED	
EE	
EF	
F0	
F1	
F2	
F3	
F4	
F5	
F6	
F7	
F8	
F9	
FA	
FB	
FC	
FD	
FE	
FF	

Figure 4–4. 256-Byte RAM Memory Map (7842)

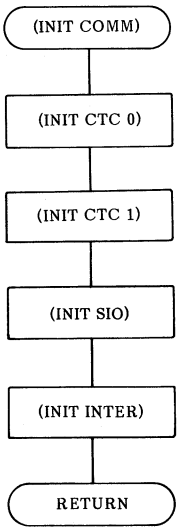


Figure 4-5. (INIT COMM) Flowchart (7842)

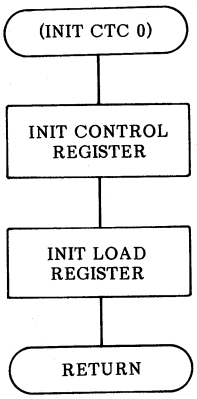


Figure 4-6. (INIT CTC 0) Flowchart (7842)

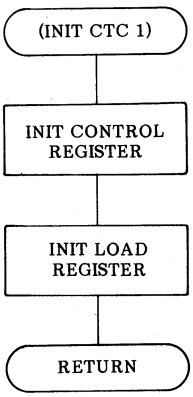


Figure 4-7. (INIT CTC 1) Flowchart (7842)

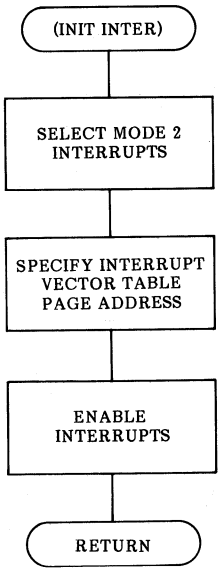


Figure 4-8. (INIT INTER) Flowchart (7842)

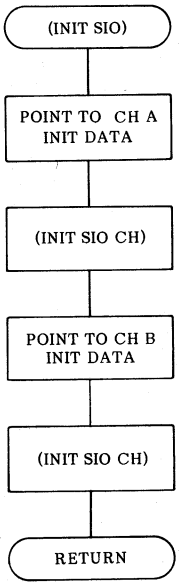


Figure 4-9. (INIT SIO) Flowchart (7842)

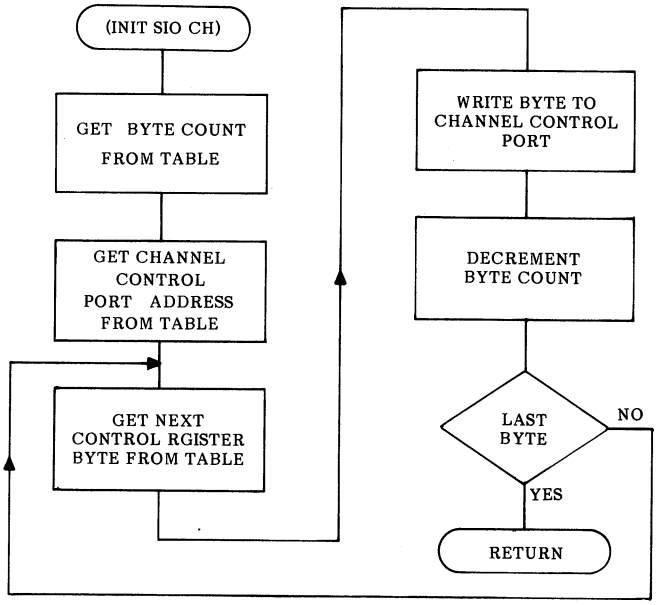


Figure 4-10. (INIT SIO CH) Flowchart (7842)

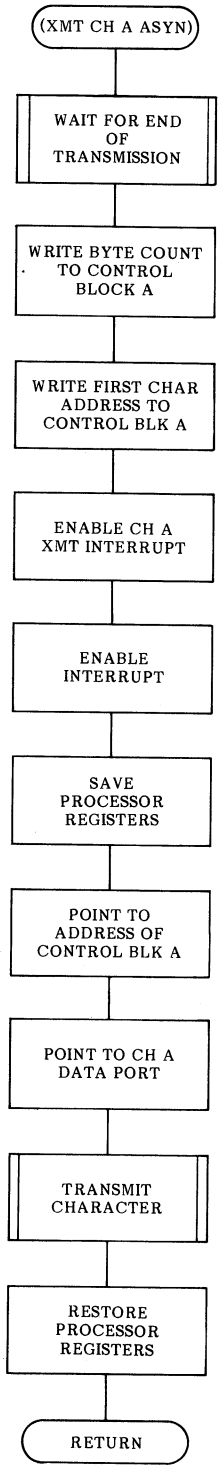

(XMT CH A ASYN)
WAIT FOR END OF TRANSMISSION
WRITE BYTE COUNT TO CONTROL BLOCK A
WRITE FIRST CHAR ADDRESS TO CONTROL BLK A
ENABLE CH A XMT INTERRUPT
ENABLE INTERRUPT
SAVE PROCESSOR REGISTERS
POINT TO ADDRESS OF CONTROL BLK A
POINT TO CH A DATA PORT
TRANSMIT CHARACTER
RESTORE PROCESSOR REGISTERS
RETURN

Figure 4-11.
(XMT CH A ASYN) Flowchart (7842)

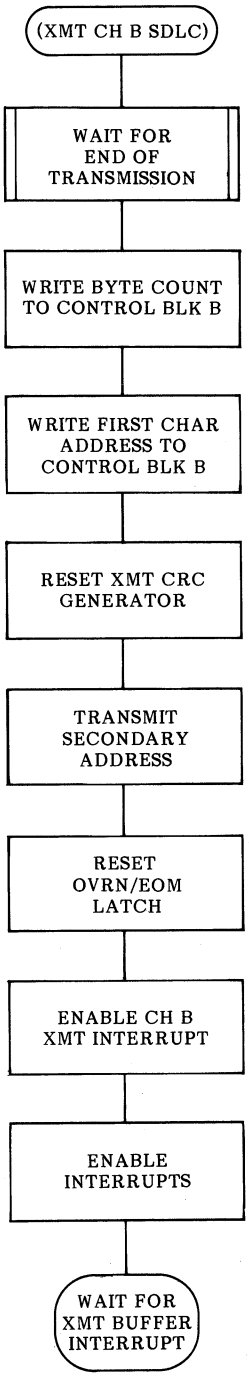

(XMT CH B SDLC)
WAIT FOR END OF TRANSMISSION
WRITE BYTE COUNT TO CONTROL BLK B
WRITE FIRST CHAR ADDRESS TO CONTROL BLK B
RESET XMT CRC GENERATOR
TRANSMIT SECONDARY ADDRESS
RESET OVRN/EOM LATCH
ENABLE CH B XMT INTERRUPT
ENABLE INTERRUPTS
WAIT FOR XMT BUFFER INTERRUPT

Figure 4-12. (XMT CH B SDLC) Flowchart (7842)

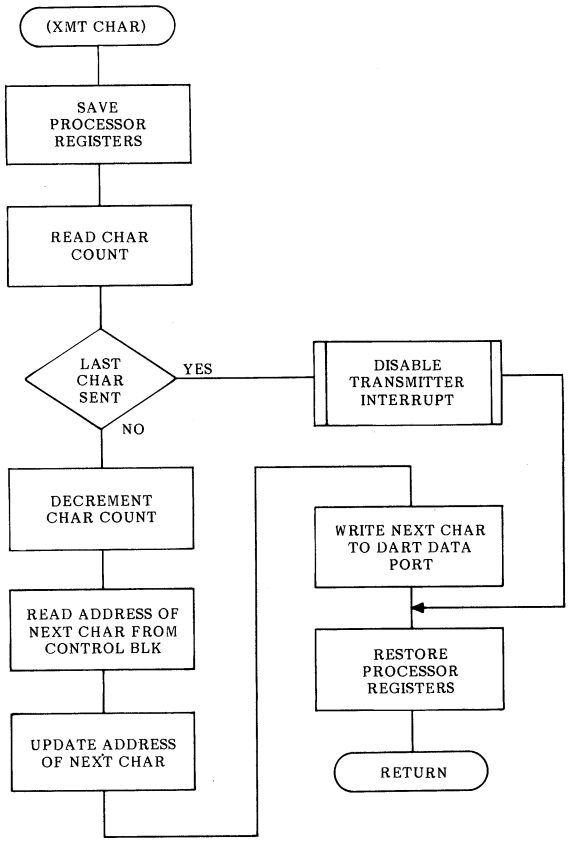


Figure 4-13. (XMT CHAR) Flowchart (7842)

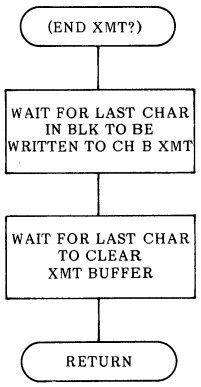


Figure 4-14. (END XMT?) Flowchart (7842)

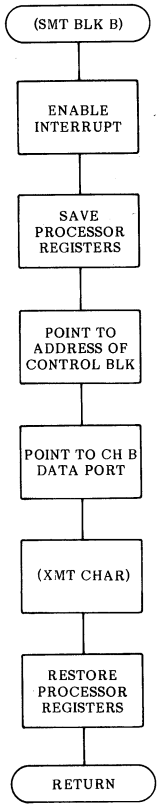


Figure 4-15. (XMT BLK B) Flowchart (7842)

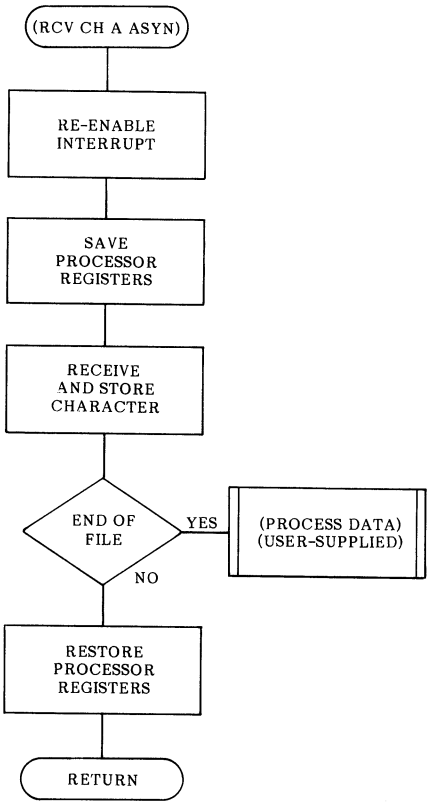


Figure 4-16. (RCV CH A ASYN) Flowchart (7842)

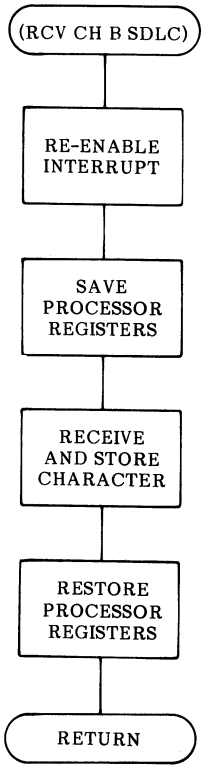


Figure 4-17. (RCV CH B SDLC) Flowchart (7842)

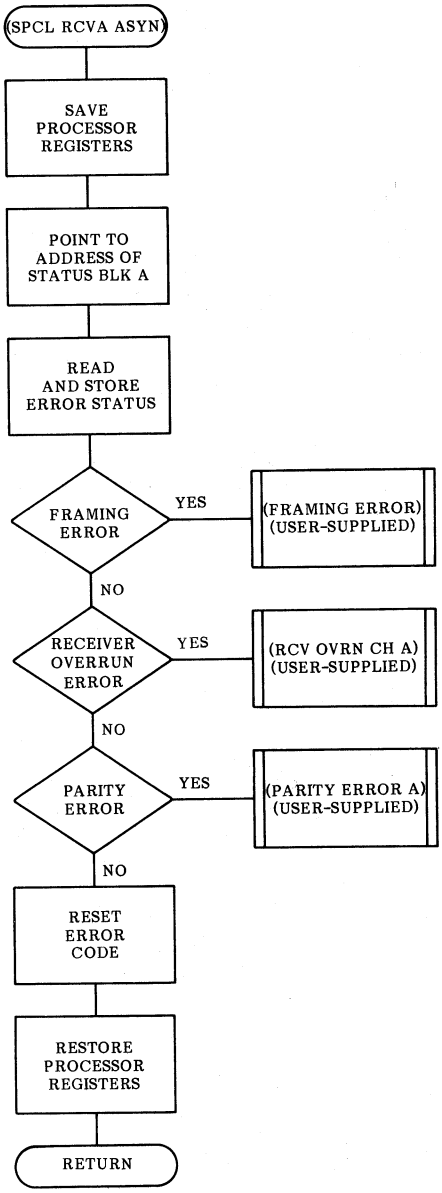


Figure 4-18. (SPCL RCV A ASYN) Flowchart (7842)

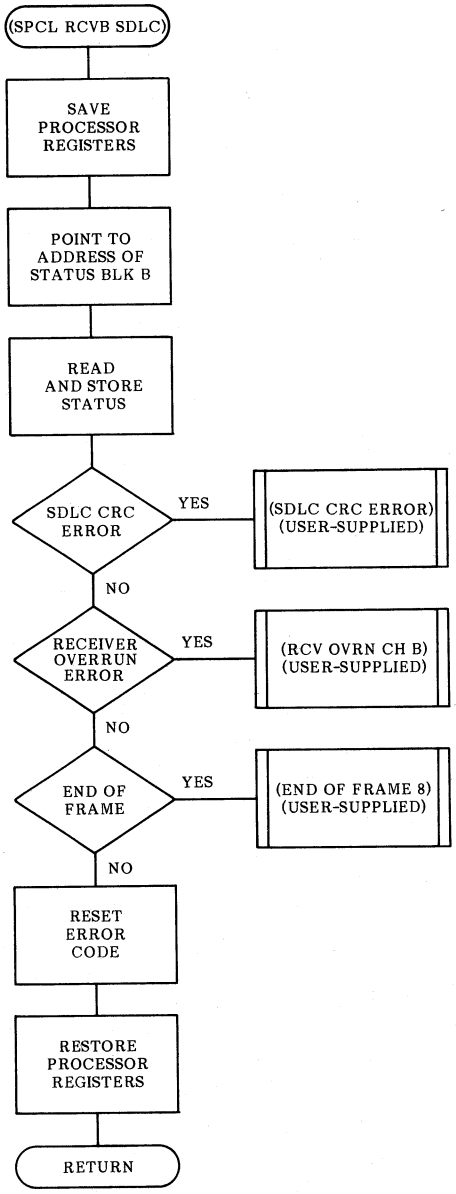


Figure 4-19. (SPCL RCV B SDLC) Flowchart (7842)

```
ADRdata!     LABEL       !INSTR !    MODIFIER    !           COMMENTS
-------!-----------------!------!----------------!-------------------------------
1D00 CD!(INIT COMM)      !JS    !UN              !INITIALIZE CTC CHANNEL 0
  01 20!                 ! ---- !(INIT CTC 0)     !  BAUD RATE 9600 CH A
  02 1D!                 ! ---- !                 !
  03 CD!                 !JS    !UN              !INITIALIZE CTC CHANNEL 1
  04 30!                 ! ---- !(INIT CTC 1)     !  BAUD RATE 100K CH B
  05 1D!                 ! ---- !                 !
  06 CD!                 !JS    !UN              !INITIALIZE SIO CHANNELS A & B
  07 40!                 ! ---- !(INIT SIO)       !
  08 1D!                 ! ---- !                 !
  09 CD!                 !JS    !UN              !INITIALIZE INTERRUPT STRUCTURE
  0A 18!                 ! ---- !(INIT INTER)     !
  0B 1D!                 ! ---- !                 !
  0C 3E!                 !LDAI  !                 !INITIALIZE SIO CONTROL BLOCK
  0D 00!                 ! ---- !00               !  COUNTS AT 00
  0E 32!                 !STAD  !                 !
  0F 00!                 ! ---- !CH A CNTRL BLK   !
1D10 32!                 ! ---- !                 !
  11 32!                 !STAD  !                 !
  12 03!                 ! ---- !CH B CNTRL BLK   !
  13 32!                 ! ---- !                 !
  14 C9!   ----------    !RTS   !UN              !RETURN
  15 FF!                 !      !                 !
  16 FF!                 !      !                 !
  17 FF!                 !      !                 !
1D18 ED!(INIT INTER)     ! SEIM !2               !SELECT MODE 2 INTERRUPTS
  19 5E!                 ! ---- !                 !
  1A 3E!                 !LDAI  !                 !SPECIFY INTERRUPT VECTOR TABLE
  1B FF!                 ! ---- !FF              !  PAGE ADDRESS (USER SUPPLIED)
  1C ED!                 !LDR   !I,A             !
  1D 47!                 ! ---- !                 !
  1E FB!                 !ENI   !                 !ENABLE INTERRUPTS
  1F C9!   ----------    !RTS   !UN              !
1D20 3E!(INIT CTC 0)     !LDAI  !                 !WRITE CHANNEL CONTROL BYTE
  21 45!                 ! ---- !45              !
  22 D3!                 !OPA   !                 !
  23 F0!                 ! ---- !F0              !
  24 3E!                 !LDAI  !                 !WRITE TIME CONSTANT BYTE
  25 0D!                 ! ---- !0D              !
  26 D3!                 !OPA   !                 !
  27 F0!                 ! ---- !F0              !
  28 C9!   ----------    !RTS   !UN              !(INITIALIZED FOR 4800 BAUD)
  29 FF!                 !      !                 !
  2A FF!                 !      !                 !
  2B FF!                 !      !                 !
  2C FF!                 !      !                 !
  2D FF!                 !      !                 !
  2E FF!                 !      !                 !
  2F FF!                 !      !                 !
1D30 3E!(INIT CTC 1)     !LDAI  !                 !WRITE CHANNEL CONTROL BYTE
  31 45!                 ! ---- !45              !
  32 D3!                 !OPA   !                 !
  33 F1!                 ! ---- !F1              !
  34 3E!                 !LDAI  !                 !WRITE TIME CONSTANT BYTE
  35 34!                 ! ---- !34              !
  36 D3!                 !OPA   !                 !
  37 F1!                 ! ---- !F1              !
  38 C9!   ----------    !RTS   !UN              !(INITIALIZED FOR 19.2KBAUD)
  39 FF!                 !      !                 !
  3A FF!                 !      !                 !
  3B FF!                 !      !                 !
  3C FF!                 !      !                 !
  3D FF!                 !      !                 !
  3E FF!                 !      !                 !
  3F FF!                 !      !                 !
```

```
ADRdata!      LABEL       !INSTR !    MODIFIER    !          COMMENTS
-------!-----------------!------!----------------!-----------------------------
1D40 21!(INIT SIO)        !LDPI  !HL              !INITIALIZE SIO CHANNEL A FOR
  41 60!                  ! ---- !1D60            !RS-232-C ASYNC (HL = 1D60),
  42 1D!                  ! ---- !                !OR RS-422 ASYNC (HL = 1D70)
  43 CD!                  !JS    !UN              !
  44 50!                  ! ---- !(INIT SIO CH)   !
  45 1D!                  ! ---- !                !
  46 21!                  !LDPI  !HL              !INITIALIZE SIO CHANNEL B FOR
  47 80!                  ! ---- !1D80            !RS-422 SDLC COMMUNICATIONS
  48 1D!                  ! ---- !                !
  49 CD!                  !JS    !UN              !
  4A 50!                  ! ---- !(INIT SIO CH)   !
  4B 1D!                  ! ---- !                !
  4C C9!   ----------     !RTS   !UN              !RETURN
  4D FF!                  !      !                !
  4E FF!                  !      !                !
  4F FF!                  !      !                !
1D50 46!(INIT SIO CH)     !LDRN  !B,(HL)          !GET NUMBER OF BYTES TO OUTPUT
  51 23!                  !ICP   !HL              !GET CHAN CONTROL PORT ADDRESS
  52 4E!                  !LDRN  !C,(HL)          !
1D53 23!NEXT BYTE         !ICP   !HL              !OUTPUT TABLE TO CHANNEL CONTROL
  54 7E!                  !LDAN  !(HL)            !  FIRST SELECTING THE WRITE REG
  55 ED!                  !OPAN  !(C)             !  AND THEN OUTPUTTING ITS
  56 79!                  ! ---- !                !  CONTROL BYTE
  57 05!                  !DCR   !B               !
  58 C2!                  !JP    !NOT ZERO        !
  59 53!                  ! ---- !NEXT BYTE       !
  5A 1D!                  ! ---- !                !
  5B C9!   ----------     !RTS   !UN              !
  5C FF!                  !      !                !
  5D FF!                  !      !                !
  5E FF!                  !      !                !
  5F FF!                  !      !                !
1D60 0A!CHA ASY 232 DATA  !  0A  !                !NUMBER OF BYTES TO OUTPUT
  61 F5!                  !  F5  !                !CHANNEL A CONTROL PORT
  62 18!                  !  18  !                !CHANNEL A RESET
  63 04!                  !  04  !                !WR4A
  64 47!                  !  47  !                !X16 CLK,1 STOP B/C, EVEN PARITY
  65 03!                  !  03  !                !WR3A
  66 E1!                  !  E1  !                !RCV 8B/C,AUTO EN
  67 05!                  !  05  !                !WR5A
  68 EA!                  !  EA  !                !DTR*,XMT 8B/C,XMT EN,RTS*
  69 01!                  !  01  !                !WR1A
1D6A 1C!adr 1D6A          !  1C  !                !INT ALL RCV CHAR,STAT AFF VECTR
  6B 30!                  !  30  !                !ERROR RESET
  6C FF!                  !      !                !
  6D FF!                  !      !                !
  6E FF!                  !      !                !
  6F FF!                  !      !                !
1D70 0A!CHA ASY 422 DATA  !  0A  !                !NUMBER OF BYTES TO OUTPUT
  71 F5!                  !  F5  !                !CHANNEL A CONTROL PORT
  72 18!                  !  18  !                !CHANNEL A RESET
  73 04!                  !  04  !                !WR4A
  74 47!                  !  47  !                !X16 CLK,1 STOP B/C,EVEN PARITY
  75 03!                  !  03  !                !WR3A
  76 C1!                  !  C1  !                !RCV 8B/C,RCV ENABLE
  77 05!                  !  05  !                !WR5A
  78 6A!                  !  6A  !                !XMT 8B/C, XMT ENABLE
  79 01!                  !  01  !                !WR1A
  7A 1C!                  !  1C  !                !INT ALL RCV CHAR,STAT AFF VECTR
  7B 30!                  !  30  !                !ERROR RESET
  7C FF!                  !      !                !
  7D FF!                  !      !                !
  7E FF!                  !      !                !
  7F FF!                  !      !                !
```

ADR	data	LABEL	INSTR	MODIFIER	COMMENTS
1D80	10	CH B SDLC DATA	10		NUMBER OF BYTES TO OUTPUT
81	F7		F7		CHANNEL B CONTROL PORT
82	18		18		CHANNEL B RESET
83	02		02		WR2B
84	00		00		INTERRUPT VECTOR
85	04		04		WR4B
86	20		20		SDLC,X1 CLK, NO PARITY
87	06		06		WR6B
88	FF		FF		SECONDARY ADDR (USER SELECTED)
89	07		07		WR7B
8A	7E		7E		SDLC FLAG
8B	05		05		WR5B
8C	6B		6B		SDLC,X1 CLK,NO PARITY
8D	03		03		WR3B
8E	CD		CD		RCV EN, 8B/C, RCV CRC ENABLE
8F	01		01		WR1B
1D90	1C	adr 1D90	1C		INT ALL RCV CHAR,STAT AFF VECTR
91	30		30		ERROR RESET
92	FF				
93	FF				
94	FF				
1D95	F5	(XMT CH A ASYN)	PSP	AF	SAVE PROCESSOR REGISTER
96	CD		JS	UN	WAIT FOR END OF TRANSMISSION
97	00		----	(END XMT A?)	IN PROGRESS
98	1E		----		
99	F1		PLP	AF	
9A	32		STAD		WRITE BYTE COUNT TO
9B	00		----	CH A CNTRL BLK	CONTROL BLOCK
9C	32		----		
9D	22		STPD	HL	WRITE FIRST CHARACTER ADDRESS
9E	01		----	adr 3201	TO CONTROL BLOCK
9F	32		----		
1DA0	3E		LDAI		SELECT WR1A
A1	01		----	01	
A2	D3		OPA		
A3	F5		----	F5	
A4	3A		LDAD		
A5	6A		----	adr 1D6A	
A6	1D		----		
A7	F6		ORAI		ENABLE XMT INTERRUPT
A8	02		----	02	
A9	D3		OPA		
AA	F5		----	F5	
1DAB	FB	XMT BLK A	ENI		ENABLE INTERRUPTS
AC	C5		PSP	BC	SAVE PROCESSOR REGISTERS
AD	E5		PSP	HL	
AE	21		LDPI	HL	POINT TO ADDRESS OF
AF	00		----	3200	CONTROL BLOCK
1DB0	32		----		
B1	0E		LDRI	C	POINT TO CHANNEL A DATA PORT
B2	F4		----	F4	
B3	CD		JS	UN	TRANSMIT CHARACTER
B4	20		----	(XMT CHAR)	
B5	1E		----		
B6	E1		PLP	HL	RESTORE PROCESSOR REGISTERS
B7	C1		PLP	BC	
B8	ED		RTI	UN	RETURN
B9	4D	--------	----		
BA	FF				
BB	FF				
BC	FF				
BD	FF				
BE	FF				
BF	FF				

ADR	data	LABEL	INSTR	MODIFIER	COMMENTS
1DC0	F5	(XMT CH B SDLC)	PSP	AF	SAVE PROCESSOR REGISTER
C1	CD		JS	UN	WAIT FOR END OF TRANSMISSION
C2	10		----	(END XMT B?)	IN PROGRESS
C3	1E		----		
C4	F1		PLP	AF	
C5	32		STAD		WRITE BYTE COUNT TO
C6	03		----	CH B CNTRL BLK	CONTROL BLOCK B
C7	32		----		
C8	22		STPD	HL	WRITE FIRST CHARACTER ADDRESS
C9	04		----	adr 3204	TO CONTROL BLOCK B
CA	32		----		
CB	3E		LDAI		RESET XMT CRC GENERATOR
CC	80		----	80	
CD	D3		OPA		
CE	F7		----	F7	
CF	3E		LDAI		TRANSMIT SECONDARY ADDRESS
1DD0	FF		----	FF	BYTE (USER SELECTED)
D1	D3		OPA		
D2	F6		----	F6	
D3	3E		LDAI		RESET OVRN/EOM LATCH AND
D4	C1		----	C1	POINT TO WR1B
D5	D3		OPA		
D6	F7		----	F7	
D7	3A		LDAD		
D8	90		----	adr 1D90	
D9	1D		----		
DA	F6		ORAI		ENABLE TRANSMIT INTERRUPTS
DB	02		----	02	
DC	D3		OPA		
DD	F7		----	F7	
DE	FB		ENI		ENABLE INTERRUPTS
1DDF	C3	WAIT	JP	UN	WAIT FOR TRANSMIT BUFFER
1DE0	DF		----	WAIT	TO EMPTY
E1	1D	----------	----		
1DE2	FB	(XMT BLK B)	ENI		ENABLE INTERRUPTS
E3	C5		PSP	BC	SAVE PROCESSOR REGISTERS
E4	E5		PSP	HL	
E5	21		LDPI	HL	POINT TO ADDRESS OF
E6	03		----	3203	CONTROL BLOCK B
E7	32		----		
E8	0E		LDRI	C	POINT TO CHANNEL B DATA PORT
E9	F6		----	F6	
EA	CD		JS	UN	TRANSMIT CHARACTER
EB	20		----	(XMT CHAR)	
EC	1E		----		
ED	E1		PLP	HL	RESTORE PROCESSOR REGISTERS
EE	C1		PLP	BC	
EF	ED		RTI	UN	RETURN
1DF0	4D	---------	----		
F1	FF				
F2	FF				
F3	FF				
F4	FF				
F5	FF				
F6	FF				
F7	FF				
F8	FF				
F9	FF				
FA	FF				
FB	FF				
FC	FF				
FD	FF				
FE	FF				
FF	FF				

```
ADRdata!      LABEL       !INSTR !    MODIFIER    !            COMMENTS
-------!-----------------!------!----------------!-----------------------------
1E00 3A!(END XMT A?)      !LDAD  !                !WAIT FOR LAST CHARACTER IN
  01 00!                  ! ---- !CH A CNTRL BLK  !  BLOCK TO BE WRITTEN TO
  02 32!                  ! ---- !                !  CHANNEL A
  03 FE!                  !CPAI  !                !
  04 00!                  ! ---- !00              !
  05 C2!                  !JP    !NOT ZERO        !
  06 00!                  ! ---- !(END XMT A?)    !
  07 1E!                  ! ---- !                !
1E08 DB!BUFF EMPTY* A     !IPA   !                !WAIT FOR LAST CHARACTER
  09 F5!                  ! ---- !F5              !  TO CLEAR XMT BUFFER
  0A E6!                  !ANAI  !                !
  0B 04!                  ! ---- !04              !
  0C CA!                  !JP    !ZERO            !
  0D 08!                  ! ---- !BUFF EMPTY* A   !
  0E 1E!                  ! ---- !                !
  0F C9!   ----------     !RTS   !UN              !
1E10 3A!(END XMT B?)      !LDAD  !                !WAIT FOR LAST CHARACTER IN
  11 03!                  ! ---- !CH B CNTRL BLK  !  BLOCK TO BE WRITTEN TO
  12 32!                  ! ---- !                !  CHANNEL B XMT BUFFER
  13 FE!                  !CPAI  !                !
  14 00!                  ! ---- !00              !
  15 C2!                  !JP    !NOT ZERO        !
  16 10!                  ! ---- !(END XMT B?)    !
  17 1E!                  ! ---- !                !
1E18 DB!BUFF EMPTY* B     !IPA   !                !WAIT FOR LAST CHARACTER TO
  19 F7!                  ! ---- !F7              !  CLEAR XMT BUFFER
  1A E6!                  !ANAI  !                !
  1B 04!                  ! ---- !04              !
  1C CA!                  !JP    !ZERO            !
  1D 18!                  ! ---- !BUFF EMPTY* B   !
  1E 1E!                  ! ---- !                !
  1F C9!   ----------     !RTS   !UN              !
1E20 F5!(XMT CHAR)        !PSP   !AF              !SAVE PROCESSOR REGISTERS
  21 D5!                  !PSP   !DE              !
  22 7E!                  !LDAN  !(HL)            !HAS LAST CHARACTER BEEN SENT?
  23 FE!                  !CPAI  !                !
  24 00!                  ! ---- !00              !
  25 CA!                  !JP    !ZERO            !
  26 38!                  ! ---- !LAST            !
  27 1E!                  ! ---- !                !
  28 3D!                  !DCA   !                !NO, DECREMENT COUNT & STORE
  29 77!                  !STAN  !(HL)            !
  2A 23!                  !ICP   !HL              !READ ADDRESS OF NEXT CHARACTER
  2B 5E!                  !LDRN  !E,(HL)          !  FROM CONTROL BLOCK
  2C 23!                  !ICP   !HL              !
  2D 56!                  !LDRN  !D,(HL)          !
  2E 1A!                  !LDAN  !(DE)            !READ NEXT CHARACTER
  2F 13!                  !ICP   !DE              !UPDATE ADDRESS OF NEXT
1E30 72!                  !STRN  !(HL),D          !  CHARACTER AND STORE IN
  31 2B!                  !DCP   !HL              !  CONTROL BLOCK
  32 73!                  !STRN  !(HL),E          !
  33 ED!                  !OPAN  !(C)             !WRITE TO SIO DATA PORT
  34 79!                  ! ---- !                !
  35 D1!                  !PLP   !DE              !RESTORE PROCESSOR REGISTERS
  36 F1!                  !PLP   !AF              !
  37 C9!   ----------     !RTS   !UN              !RETURN
1E38 0C!LAST              !ICR   !C               !LAST CHARACTER SENT, TURN OFF
  39 3E!                  !LDAI  !                !  XMT INTERRUPT
  3A 28!                  ! ---- !28              !
  3B ED!                  !OPAN  !(C)             !
  3C 79!                  ! ---- !                !
  3D D1!                  !PLP   !DE              !RESTORE PROCESSOR REGISTERS
  3E F1!                  !PLP   !AF              !
  3F C9!   ----------     !RTS   !UN              !RETURN
```

ADR	data	LABEL	INSTR	MODIFIER	COMMENTS
1E40	FB	(RCV CH A ASYN)	ENI		ENABLE INTERRUPTS
41	F5		PSP	AF	SAVE PROCESSOR REGISTERS
42	DB		IPA		READ CHARACTER FROM SIO
43	F4		----	F4	CHANNEL A DATA PORT
44	77		STAN	(HL)	STORE DATA IN MEMORY
45	23		ICP	HL	
46	FE		CPAI		0A = ASCII NEW LINE CHARACTER
47	0A		----	0A	
48	CA		JS	ZERO	WHEN END OF FILE RECEIVED
49	--		----	(PROC DATA)	PROCESS DATA IN USER-
4A	--		----		SUPPLIED ROUTINE
4B	F1		PLP	AF	
4C	ED		RTI	UN	
4D	4D	--------	----		RETURN
4E	FF				
4F	FF				
1E50	FB	(RCV CH B SDLC)	ENI		ENABLE INTERRUPTS
51	F5		PSP	AF	SAVE PROCESSOR REGISTERS
52	DB		IPA		READ CHARACTER FROM SIO
53	F6		----	F6	CHANNEL B DATA PORT
54	77		STAN	(HL)	STORE DATA IN MEMORY
55	23		ICP	HL	
56	F1		PLP	AF	RESTORE PROCESSOR REGISTERS
57	ED		RTI	UN	
58	4D	--------	----		RETURN
59	FF				
5A	FF				
1E5B	F5	(SPCL RCVA ASYN)	PSP	AF	SAVE PROCESSOR REGISTERS
5C	E5		PSP	HL	
5D	21		LDPI	HL	POINT TO ADDRESS OF CHANNEL A
5E	06		----	3206	STATUS BLOCK
5F	32		----		
1E60	3E		LDAI		READ AND STORE STATUS RR1A
61	01		----	01	
62	D3		OPA		
63	F5		----	F5	
64	DB		IPA		
65	F5		----	F5	
66	77		STAN	(HL)	
67	E6		ANAI		IF FRAMING ERROR OCCURS,
68	40		----	40	GO TO USER-SUPPLIED
69	C4		JS	NOT ZERO	ERROR ROUTINE
6A	--		----	(FRAMING ERROR)	
6B	--		----		
6C	7E		LDAN	(HL)	
6D	E6		ANAI		IF RECEIVER OVERRUN ERROR
6E	20		----	20	OCCURS, GO TO USER-SUPPLIED
6F	C4		JS	NOT ZERO	ERROR ROUTINE
1E70	--		----	(RCV OVRN CH A)	
71	--		----		
72	7E		LDAN	(HL)	
73	E6		ANAI		IF PARITY ERROR OCCURS,
74	10		----	10	GO TO USER-SUPPLIED
75	C4		JS	NOT ZERO	ERROR ROUTINE
76	--		----	(PARITY ERROR A)	
77	--		----		
78	3E		LDAI		RESET ERROR CODE
79	30		----	30	
7A	D3		OPA		
7B	F5		----	F5	
7C	E1		PLP	HL	RESTORE PROCESSOR REGISTERS
7D	F1		PLP	AF	
7E	ED		RTI	UN	RETURN
7F	4D	--------	----		

```
ADRdata!      LABEL       !INSTR !    MODIFIER    !          COMMENTS
-------!-----------------!------!----------------!-----------------------------
1E80 F5!(SPCL RCVB SDLC)!PSP   !AF              !SAVE PROCESSOR REGISTERS
  81 E5!                 !PSP   !HL              !
  82 21!                 !LDPI  !HL              !POINT TO ADDRESS OF CHANNEL B
  83 07!                 ! ---- !3207            !  STATUS BLOCK
  84 32!                 ! ---- !                !
  85 3E!                 !LDAI  !                !READ AND STORE STATUS RR1B
  86 01!                 ! ---- !01              !
  87 D3!                 !OPA   !                !
  88 F7!                 ! ---- !F7              !
  89 DB!                 !IPA   !                !
  8A F7!                 ! ---- !F7              !
  8B 77!                 !STAN  !(HL)            !
  8C E6!                 !ANAI  !                !IF SDLC CRC ERROR OCCURS
  8D 40!                 ! ---- !40              !  GOT TO USER-SUPPLIED
  8E C4!                 !JS    !NOT ZERO        !  ERROR-ROUTINE
  8F --!                 ! ---- !(SDLC CRC ERROR)!
1E90 --!                 ! ---- !                !
  91 7E!                 !LDAN  !(HL)            !
  92 E6!                 !ANAI  !                !IF RECEIVER OVERRUN ERROR
  93 20!                 ! ---- !20              !  OCCURS, GO TO USER-SUPPLIED
  94 C4!                 !JS    !NOT ZERO        !  ERROR ROUTINE
  95 --!                 ! ---- !(RCV OVRN CH B) !
  96 --!                 ! ---- !                !
  97 7E!                 !LDAN  !(HL)            !
  98 E6!                 !ANAI  !                !IF END OF FRAME IS DETECTED,
  99 80!                 ! ---- !80              !  GO TO USER-SUPPLIED ROUTINE
  9A C4!                 !JS    !NOT ZERO        !
  9B --!                 ! ---- !(END OF FRAME B)!
  9C --!                 ! ---- !                !
  9D 3E!                 !LDAI  !                !RESET ERROR CODE
  9E 30!                 ! ---- !30              !
  9F D3!                 !OPA   !                !
1EA0 F5!                 ! ---- !F5              !
  A1 E1!                 !PLP   !HL              !RESTORE PROCESSOR REGISTERS
  A2 F1!                 !PLP   !AF              !
  A3 ED!                 !RTI   !UN              !
  A4 4D! ---------       !----  !                !RETURN
  A5 FF!                 !      !                !
  A6 FF!                 !      !                !
  A7 FF!                 !      !                !
  A8 FF!                 !      !                !
  A9 FF!                 !      !                !
  AA FF!                 !      !                !
  AB FF!                 !      !                !
  AC FF!                 !      !                !
  AD FF!                 !      !                !
  AE FF!                 !      !                !
  AF FF!                 !      !                !
1EB0 FF!                 !      !                !
  B1 FF!                 !      !                !
  B2 FF!                 !      !                !
  B3 FF!                 !      !                !
  B4 FF!                 !      !                !
  B5 FF!                 !      !                !
  B6 FF!                 !      !                !
  B7 FF!                 !      !                !
  B8 FF!                 !      !                !
  B9 FF!                 !      !                !
  BA FF!                 !      !                !
  BB FF!                 !      !                !
  BC FF!                 !      !                !
  BD FF!                 !      !                !
  BE FF!                 !      !                !
  BF FF!                 !      !                !
```

ADRdata	LABEL	INSTR	MODIFIER	COMMENTS
3200 FF	CH A CNTRL BLK			NUMBER OF CHARACTERS IN BLOCK
01 FF				ADDRESS OF FIRST CHARACTER
02 FF				
3203 FF	CH B CNTRL BLK			NUMBER OF CHARACTERS IN BLOCK
04 FF				ADDRESS OF FIRST CHARACTER
05 FF				
3206 FF	CH A RR1			CHANNEL A ERROR STATUS
3207 FF	CH B RR1			CHANNEL B ERROR STATUS
08 FF				
09 FF				
0A FF				
0B FF				
0C FF				
0D FF				
0E FF				
0F FF				
3210 FF				
11 FF				
12 FF				
13 FF				
14 FF				
15 FF				
16 FF				
17 FF				
18 FF				
19 FF				
1A FF				
1B FF				
1C FF				
1D FF				
1E FF				
1F FF				
3220 FF				
21 FF				
22 FF				
23 FF				
24 FF				
25 FF				
26 FF				
27 FF				
28 FF				
29 FF				
2A FF				
2B FF				
2C FF				
2D FF				
2E FF				
2F FF				
3230 FF				
31 FF				
32 FF				
33 FF				
34 FF				
35 FF				
36 FF				
37 FF				
38 FF				
39 FF				
3A FF				
3B FF				
3C FF				
3D FF				
3E FF				
3F FF				

ADRdata	LABEL	INSTR	MODIFIER	COMMENTS
FF00 E2	INTERRUPT VECTOR	E2		SIO CH B SERVICE ROUTINES
01 1D		1D		TRANSMIT BUFFER EMPTY
02 FF				
03 FF				
04 50		50		RECEIVE CHARACTER AVAILABLE
05 1E		1E		
06 80		80		SPECIAL RECEIVE CONDITION
07 1E		1E		
08 AB		AB		SIO CH A SERVICE ROUTINES
09 1D		1D		TRANSMIT BUFFER EMPTY
0A FF				
0B FF				
0C 40		40		RECEIVE CHARACTER AVAILABLE
0D 1E		1E		
0E 5B		5B		SPECIAL RECEIVE CONDITION
0F 1E		1E		
FF10 FF				
11 FF				
12 FF				
13 FF				
14 FF				
15 FF				
16 FF				
17 FF				
18 FF				
19 FF				
1A FF				
1B FF				
1C FF				
1D FF				
1E FF				
1F FF				
FF20 FF				
21 FF				
22 FF				
23 FF				
24 FF				
25 FF				
26 FF				
27 FF				
28 FF				
29 FF				
2A FF				
2B FF				
2C FF				
2D FF				
2E FF				
2F FF				
FF30 FF				
31 FF				
32 FF				
33 FF				
34 FF				
35 FF				
36 FF				
37 FF				
38 FF				
39 FF				
3A FF				
3B FF				
3C FF				
3D FF				
3E FF				
3F FF				

REFERENCE DRAWINGS

The schematic (figures 5-1) and assembly drawing (figure 5-2) in the following pages are included in this manual FOR REFERENCE ONLY. Because improvements to the card may have been made since this manual was published, they may differ in some respects from the card and documentation that the user received from Pro-Log.

The schematic and assembly drawing shipped by Pro-Log with the card are those from which the card was manufactured.

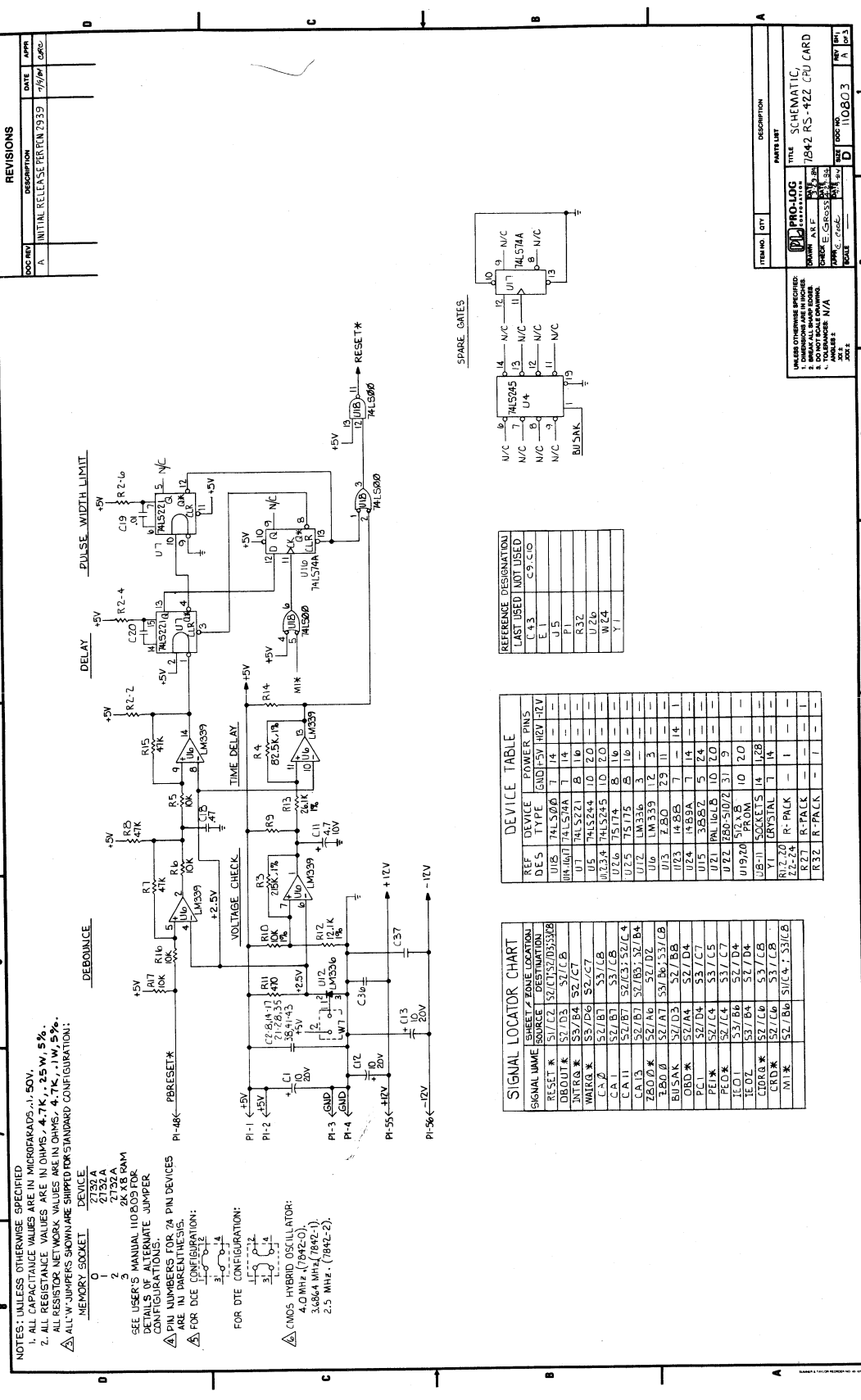

NOTES: UNLESS OTHERWISE SPECIFIED
1. ALL CAPACITANCE VALUES ARE IN MICROFARADS, ±.1, 50V.
2. ALL RESISTANCE VALUES ARE IN OHMS, 4.7K, .25W, 5%.
ALL RESISTOR NETWORK VALUES ARE IN OHMS, 4.7K, .1W, 5%.
ALL 'W' JUMPERS SHOWN ARE SHIPPED FOR STANDARD CONFIGURATION:
MEMORY SOCKET DEVICE
0 2732A
1 2732A
2 2K X8 NVRAM
3 SEE USER'S MANUAL 110 BOOSTER
DETAILS OF ALTERNATE JUMPER
CONFIGURATIONS.
PIN NUMBERS FOR 24 PIN DEVICES
ARE IN PARENTHESIS.
FOR DCE CONFIGURATION:
FOR DTE CONFIGURATION:
CMOS HYBRID OSCILLATOR:
4.0 MHz.(7842-0).
3.6864 MHz.(7842-1).
2.5 MHz.(7842-2).
DEBOUNCE
VOLTAGE CHECK
TIME DELAY
DELAY
PULSE WIDTH LIMIT
SPARE GATES
BUSAK
RESET*
PBRESET*
+5V
+12V
-12V
+2.5V
GND
P1-48
P1-1
P1-2
P1-3
P1-4
P1-55
P1-56
SIGNAL LOCATOR CHART
SIGNAL NAME | SHEET # ZONE LOCATION SOURCE | DESTINATION
RESET * | S1/C2 | S2/C7,S2/D3,S3/B8
DBOUT* | S2/D3 | S2/C8
INTRQ * | S3/B4 | S2/C7
WAITRQ * | S3/D6 | S2/C7
CA0 | S2/B7 | S3/C8
CA1 | S2/B7 | S3/C8
CA11 | S2/B7 | S2/C3; S2/C4
CA13 | S2/B7 | S2/B5; S2/B4
Z80 Ø * | S2/A6 | S2/D2
Z80 Ø | S2/A7 | S3/B6; S3/C8
BUSAK | S2/D3 | S2/B8
OBD * | S2/A4 | S2/D4
PC1 | S2/D4 | S3/C7
PE1 * | S2/C4 | S3/C5
PE0 * | S2/C4 | S3/C7
IEO1 | S3/B6 | S2/D4
IE07 | S3/B4 | S2/D4
CTORQ * | S2/C6 | S3/C8
CRD * | S2/C6 | S3/C8
M1 * | S2/B6 | S1/C4; S3/C8
DEVICE TABLE
REF DES | DEVICE TYPE | POWER PINS GND | +5V | +12V | -12V
U18 | 74LS00 | 7 | 14 | — | —
U14,16,17 | 74LS74A | 7 | 14 | — | —
U7 | 74LS221 | 8 | 16 | — | —
U5 | 74LS244 | 10 | 20 | — | —
U2,3,4 | 74LS245 | 10 | 20 | — | —
U26 | 75174 | 8 | 16 | — | —
U25 | 75175 | 8 | 16 | — | —
U12 | LM336 | 2 | 3 | — | —
U6 | LM339 | 7 | 2 | — | —
U13 | Z80 | 29 | 11 | — | —
U23 | 1488 | 7 | — | 14 | 1
U24 | 1489A | 7 | 14 | — | —
U15 | 3882 | 5 | 24 | — | —
U21 | PAL 16L8 | 10 | 20 | — | —
U22 | 280-SIO/2 | 31 | 9 | — | —
U19,20 | 512 X 8 PROM | 10 | 20 | — | —
U8-11 | SOCKETS | 14 | 28 | — | —
Y1 | CRYSTAL | 7 | 14 | — | —
R1,2,20 22-24 | R-PACK | — | 1 | — | —
R27 | R-PACK | — | — | — | 1
R32 | R-PACK | — | 1 | — | —
REFERENCE DESIGNATION
LAST USED | NOT USED
C43 | C9,C10
E1 |
J5 |
P1 |
R32 |
U26 |
Y1 |
REVISIONS
DOC REV | DESCRIPTION | DATE | APPR
A | INITIAL RELEASE PER PCN 7939 | 7/15/81 | CMC
UNLESS OTHERWISE SPECIFIED:
1. DIMENSIONS ARE IN INCHES
2. BREAK ALL SHARP EDGES
3. DO NOT SCALE DRAWING
4. TOLERANCES: N/A
FRACT ±
XX ±
.XXX ±
ITEM NO | QTY | DESCRIPTION
PARTS LIST
PRO-LOG
DRAWN AKF 5-13-81
CHECK E.Gross 5-14-81
APPR. 5-14-81
SCALE —
TITLE SCHEMATIC, 7842 RS-422 CPU CARD
SIZE D | DOC NO 110803 | REV A | SH 1 OF 3

Figure 5-1. Schematic (7842), page 1
FOR REFERENCE ONLY

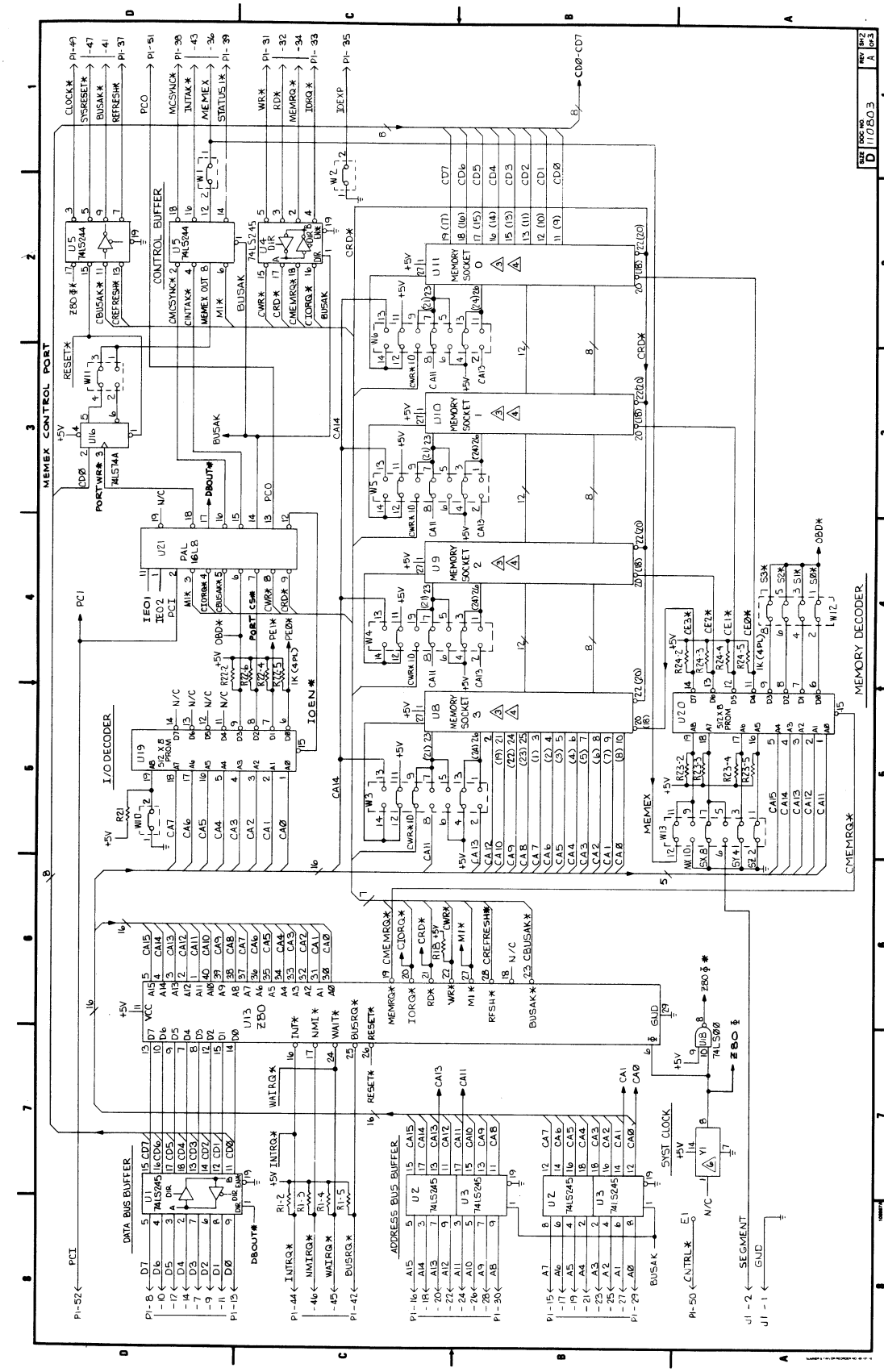


Figure 5-1 (continued). **Schematic (7842), page 2**
FOR REFERENCE ONLY

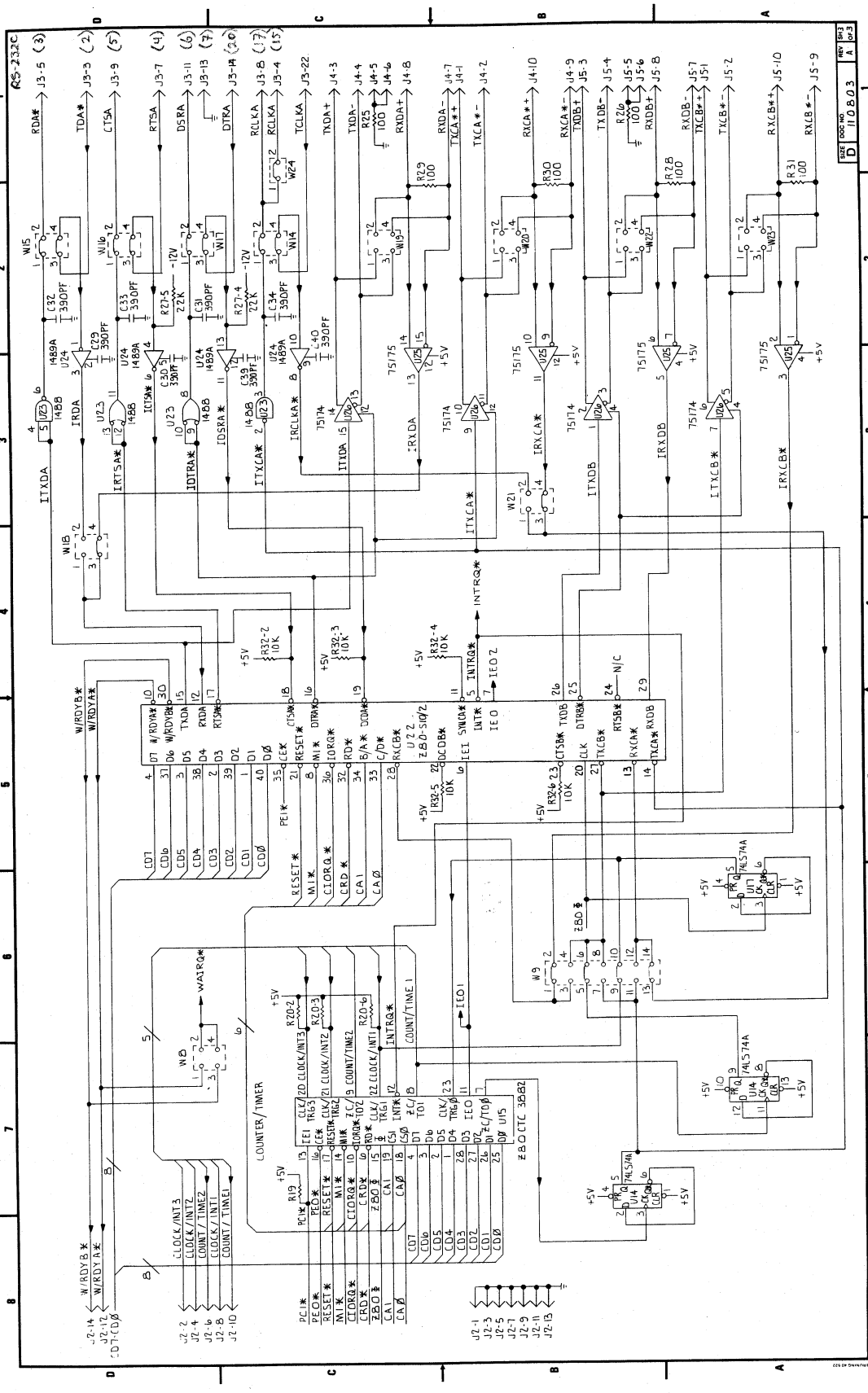


Figure 5-1 (continued). **Schematic (7842), page 3**
FOR REFERENCE ONLY

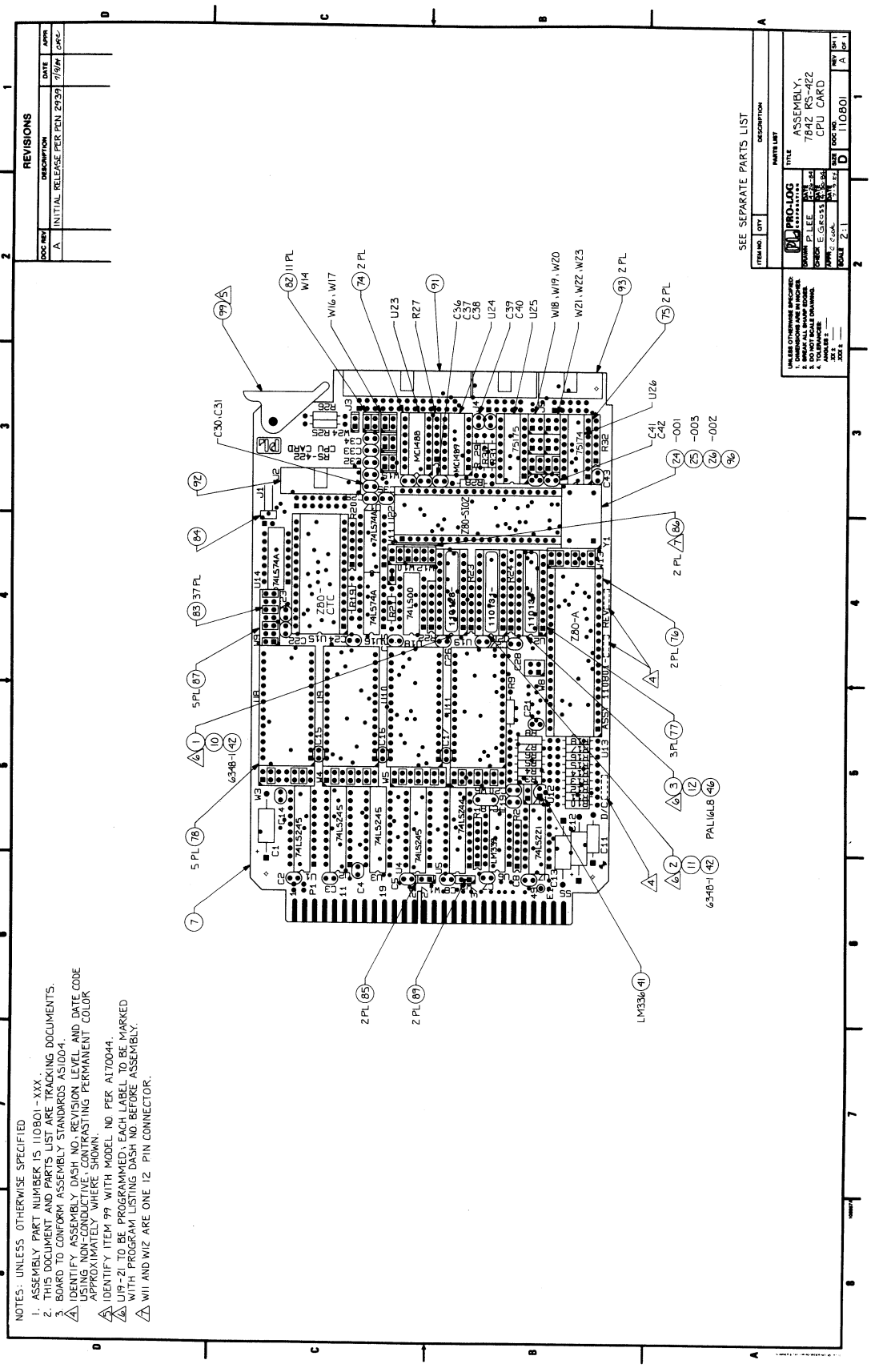


Figure 5-2. Assembly Drawing (7842)
FOR REFERENCE ONLY

RETURN FOR REPAIR PROCEDURES

Domestic Customers:

1. Call CUSTOMER SERVICE at (408) 372-4593 (in California) or (800) 538-9570 (outside of California) 7 AM to 3 PM PST.

2. Explain the problem and we may be able to solve it on the phone. If not, we will give you a Customer Return Order (CRO) number.

   Mark the CRO number on the shipping label, packing slip, and other paperwork accompanying the return. We cannot accept returns without a CRO.

3. Please be sure to enclose a packing slip with the CRO number, the equipment serial number if applicable, and the reason for return. Also enclose the name and telephone number of the person we should contact (preferably the user).

4. Package the equipment in a solid cardboard box secured with packing material.

   CAUTION: Loose MOS integrated circuits, or any product containing CMOS integrated circuits, must be protected from electrostatic discharge during handling. Use conductive foam pads or conductive plastic bags, and never place MOS or CMOS circuitry in contact with Styrofoam materials.

5. Ship prepaid and insured to:

   Pro-Log Corporation
   2411 Garden Road
   Monterey, CA 93940
   Reference CRO #_____.

International Customers:

Equipment repair is handled by your local Pro-Log distributor. If you need to contact Pro-Log, the factory can be reached at any time by TWX at 910-360-7082.

LIMITED WARRANTY:

Seller warrants that the articles furnished hereunder are free from defects in material and workmanship and perform to applicable, published Pro-Log specifications for two years from date of shipment. This warranty is in lieu of any other warranty expressed or implied. In no event will Seller be liable for special or consequential damages as a result of any alleged breach of this warranty provision. This liability of Seller hereunder shall be limited to replacing or repairing, at its option, any defective units which are returned F.O.B. Seller's plant. Equipment or parts which have been subject to abuse, misuse, accident, alteration, neglect, unauthorized repair, or installation are not covered by warranty. Seller shall have the right of final determination as to the existence and cause of defect. As to items repaired or replaced, the warranty shall continue in effect for the remainder of the warranty period, or for ninety (90) days following date of shipment by Seller of the repaired or replaced part whichever period is longer. No liability is assumed for expendable items such as lamps and fuses. No warranty is made with respect to custom equipment or products produced to buyer's specifications except as specifically stated in writing by Seller and contained in the contract.

Cut Here

FOLD

FOLD

PRO-LOG® CORPORATION

Yes, I'm interested in Pro-Log.
Please send me more information about your:

- ☐ 7000 Series STD BUS Cards
- ☐ STD BUS Packaging Solutions
- ☐ STD BUS Subsystem
- ☐ STD BUS Systems
- ☐ Designer Software
- ☐ Design Courses and Seminars
- ☐ Programmer Product Line
- ☐ Other _____

- ☐ I am currently using Pro-Log equipment.
- ☐ Add my name to your mailing list.
- ☐ Please have a Sales Representative contact me.

Reader Comments: _____

Name _____ City _____
Company _____ State _____ Zip _____
Address _____ Telephone _____

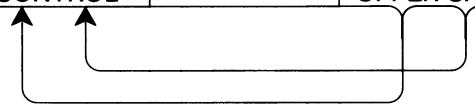
PLACE
POSTAGE
HERE

PRO-LOG®
CORPORATION
2411 GARDEN ROAD
MONTEREY, CA 93940

Attn: Literature Department

Notes

HEX				MSD	p=1	8	9	A	B	C	D	E	F
					p=0	0	1	2	3	4	5	6	7
	BITS				b8	p	p	p	p	p	p	p	p
					b7	0	0	0	0	1	1	1	1
					b6	0	0	1	1	0	0	1	1
LSD	b4	b3	b2	b5 b1		0	1	0	1	0	1	0	1
0	0	0	0	0		NUL	DLE	SP	0	@	P	`	p
1	0	0	0	1		SOH	DC1	!	1	A	Q	a	q
2	0	0	1	0		STX	DC2	"	2	B	R	b	r
3	0	0	1	1		ETX	DC3	#	3	C	S	c	s
4	0	1	0	0		EOT	DC4	$	4	D	T	d	t
5	0	1	0	1		ENQ	NAK	%	5	E	U	e	u
6	0	1	1	0		ACK	SYN	&	6	F	V	f	v
7	0	1	1	1		BEL	ETB	'	7	G	W	g	w
8	1	0	0	0		BS	CAN	(	8	H	X	h	x
9	1	0	0	1		HT	EM	)	9	I	Y	i	y
A	1	0	1	0		LF	SUB	*	:	J	Z	j	z
B	1	0	1	1		VT	ESC	+	;	K	[	k	{
C	1	1	0	0		FF	FS	,	<	L	\	l	|
D	1	1	0	1		CR	GS	—	=	M	]	m	}
E	1	1	1	0		SO	RS	.	>	N	^	n	~
F	1	1	1	1		SI	US	/	?	O	_	o	DEL
						CONTROL				UPPER CASE		LOWER CASE	



CONTROL CHARACTERS AND STANDARD KEYBOARD CONTROL SHIFT CHARACTERS

CTRL	@	NUL	Null	CTRL	K	VT	Vertical Tabulation	CTRL	V	SYN	Synchronous Idle
CTRL	A	SOH	Start of Heading	CTRL	L	FF	Form Feed	CTRL	W	ETB	End of Transmission Block
CTRL	B	STX	Start of Text	CTRL	M	CR	Carriage Return	CTRL	X	CAN	Cancel
CTRL	C	ETX	End of Text	CTRL	N	SO	Shift Out	CTRL	Y	EM	End of Medium
CTRL	D	EOT	End of Transmission	CTRL	O	SI	Shift In	CTRL	Z	SUB	Substitute
CTRL	E	ENQ	Enquiry	CTRL	P	DLE	Data Link Escape	CTRL	[	ESC	Escape
CTRL	F	ACK	Acknowledge	CTRL	Q	DC1	Device Control 1	CTRL	\	FS	File Separator
CTRL	G	BEL	Bell	CTRL	R	DC2	Device Control 2	CTRL	]	GS	Group Separator
CTRL	H	BS	Backspace	CTRL	S	DC3	Device Control 3	CTRL	^	RS	Record Separator
CTRL	I	HT	Horizontal Tabulation	CTRL	T	DC4	Device Control 4	CTRL	_	US	Unit Separator
CTRL	J	LF	Line Feed	CTRL	U	NAK	Negative Acknowledge			DEL	Delete

NOTE: The table above lists the keyboard character to be pressed with the Control key, the mnemonic for the function to be performed, and the description of the function. While the keyboard character is listed as uppercase, either uppercase or lowercase characters may be used.

ASCII Code Assignments

USER'S MANUAL

PRO-LOG®
CORPORATION

2411 Garden Road
Monterey, California 93940
Telephone: (408) 372-4593
TWX: 910-360-7082
Telex: 171879

MICROLOG

202 Flinders Street (Cnr. Frome Road)
Adelaide, South Australia 5000
Telephone 224 0744 A.H. 269 3169

110809-001B 1K 8/84